

DBMS LAB MID TERM TEST

COURSE CODE: BCSE302P

SLOT: L3 + L4

DATE: 19/6/23

Time left 0:21:43

Consider the following relational database of a college.

Student(RollNumber, StudentName, Address)

Teachers(TeacherID, TeacherName, TeachingSubject)

College(RollNumber, TeacherID)

1. Create tables identifying the primary keys and foreign keys, Insert necessary tuples into the tables. (Min 5 rows)

2. Enforce the Constraints

3. Write SQL to find

Find the name of the Students who live in Chennai

Find the name of the Teacher who teaches 'Cloud Computing' subject

Find the Name of the teacher who teaches 'Java Programming' subject to the student 'Raman'

List out the RollNo and Name who have enrolled with the teacher Name 'Radha Krishnan'

1.

-- Create Student table

```
CREATE TABLE Student (  
  RollNumber INT PRIMARY KEY,  
  StudentName VARCHAR(50),  
  Address VARCHAR(100)  
);
```

-- Create Teachers table

```
CREATE TABLE Teachers (  
  TeacherID INT PRIMARY KEY,  
  TeacherName VARCHAR(50),  
  TeachingSubject VARCHAR(50)  
);
```

-- Create College table with foreign key constraints

```
CREATE TABLE College (  
  RollNumber INT,  
  TeacherID INT,  
  PRIMARY KEY (RollNumber, TeacherID),  
  FOREIGN KEY (RollNumber) REFERENCES Student(RollNumber),  
  FOREIGN KEY (TeacherID) REFERENCES Teachers(TeacherID)  
);
```

```
SQL> CREATE TABLE Student ( RollNumber INT PRIMARY KEY, StudentName VARCHAR(50), Address VARCHAR(100));
```

Table created.

```
SQL> CREATE TABLE Teachers ( TeacherID INT PRIMARY KEY, TeacherName VARCHAR(50), TeachingSubject VARCHAR(50));
```

Table created.

```
SQL> CREATE TABLE College ( RollNumber INT, TeacherID INT, PRIMARY KEY (RollNumber, TeacherID), FOREIGN KEY (RollNumber) REFERENCES Student(RollNumber), FOREIGN KEY (TeacherID) REFERENCES Teachers(TeacherID));
```

Table created.

SQL>

-- Insert tuples into Student table

```
INSERT INTO Student (RollNumber, StudentName, Address) VALUES (1, 'Raman', 'Chennai');
INSERT INTO Student (RollNumber, StudentName, Address) VALUES (2, 'Payal', 'Mumbai');
INSERT INTO Student (RollNumber, StudentName, Address) VALUES (3, 'Neha', 'Delhi');
INSERT INTO Student (RollNumber, StudentName, Address) VALUES (4, 'Shruti', 'Chennai');
INSERT INTO Student (RollNumber, StudentName, Address) VALUES (5, 'Ramesh', 'Kolkata');
```

-- Insert tuples into Teachers table

```
INSERT INTO Teachers (TeacherID, TeacherName, TeachingSubject) VALUES (1, 'Nina Singh', 'Mathematics');
INSERT INTO Teachers (TeacherID, TeacherName, TeachingSubject) VALUES (2, 'Rithwik Dhanjani', 'English Literature');
INSERT INTO Teachers (TeacherID, TeacherName, TeachingSubject) VALUES (3, 'Karan Wahi', 'Physics');
INSERT INTO Teachers (TeacherID, TeacherName, TeachingSubject) VALUES (4, 'Sham Sharma', 'Cloud Computing');
INSERT INTO Teachers (TeacherID, TeacherName, TeachingSubject) VALUES (5, 'Radha Krishnan', 'Java Programming');
```

-- Insert tuples into College table

```
INSERT INTO College (RollNumber, TeacherID) VALUES (1, 4);
INSERT INTO College (RollNumber, TeacherID) VALUES (2, 2);
INSERT INTO College (RollNumber, TeacherID) VALUES (3, 3);
INSERT INTO College (RollNumber, TeacherID) VALUES (4, 4);
INSERT INTO College (RollNumber, TeacherID) VALUES (5, 5);
```

```
SQL> INSERT INTO Student (RollNumber, StudentName, Address) VALUES (3, 'Neha', 'Delhi');
1 row created.
SQL> INSERT INTO Student (RollNumber, StudentName, Address) VALUES (4, 'Shruti', 'Chennai');
1 row created.
SQL> INSERT INTO Student (RollNumber, StudentName, Address) VALUES (5, 'Ramesh', 'Kolkata');
1 row created.
SQL> INSERT INTO Teachers (TeacherID, TeacherName, TeachingSubject) VALUES (1, 'Nina Singh', 'Mathematics');
1 row created.
SQL> INSERT INTO Teachers (TeacherID, TeacherName, TeachingSubject) VALUES (2, 'Rithwik Dhanjani', 'English Literature');
1 row created.
SQL> INSERT INTO Teachers (TeacherID, TeacherName, TeachingSubject) VALUES (3, 'Karan Wahi', 'Physics');
1 row created.
SQL> INSERT INTO Teachers (TeacherID, TeacherName, TeachingSubject) VALUES (4, 'Sham Sharma', 'Cloud Computing');
1 row created.
SQL> INSERT INTO Teachers (TeacherID, TeacherName, TeachingSubject) VALUES (5, 'Radha Krishnan', 'Java Programming');
1 row created.
SQL> INSERT INTO College (RollNumber, TeacherID) VALUES (1, 4);
1 row created.
SQL> INSERT INTO College (RollNumber, TeacherID) VALUES (2, 2);
1 row created.
SQL> INSERT INTO College (RollNumber, TeacherID) VALUES (3, 3);
1 row created.
```

View Tables:

```
SQL> set linesize 500;
SQL> select*from Student;

ROLLNUMBER  STUDENTNAME  ADDRESS
-----
1 Raman      Chennai
2 Payal      Mumbai
3 Neha       Delhi
4 Shruti     Chennai
5 Ramesh     Kolkata

SQL> select*from Teachers;

TEACHERID  TEACHERNAME  TEACHINGSUBJECT
-----
1 Nina Singh      Mathematics
2 Rithwik Dhanjani English Literature
3 Karan Wahi      Physics
4 Sham Sharma     Cloud Computing
5 Radha Krishnan  Java Programming

SQL> select*from College;

ROLLNUMBER  TEACHERID
-----
1          4
2          2
3          3
4          4
5          5
```

2. Enforcing Constraints

```
SQL Plus

SQL> desc Student;
Name                                     Null?    Type
-----
ROLLNUMBER                               NOT NULL NUMBER(38)
STUDENTNAME                              VARCHA2(50)
ADDRESS                                  VARCHA2(100)
GENDER                                   VARCHA2(10)

SQL> desc Teachers;
Name                                     Null?    Type
-----
TEACHERID                                NOT NULL NUMBER(38)
TEACHERNAME                              VARCHA2(50)
TEACHINGSUBJECT                          VARCHA2(50)

SQL> desc College;
Name                                     Null?    Type
-----
ROLLNUMBER                               NOT NULL NUMBER(38)
TEACHERID                                NOT NULL NUMBER(38)
```

-- Alter table to add TeachingSubject as unique and allow NULL

ALTER TABLE Teachers

ADD CONSTRAINT uq_Teaching_Subject UNIQUE (TeachingSubject);

```
SQL> ALTER TABLE Teachers
2 ADD CONSTRAINT uq_Teaching_Subject UNIQUE (TeachingSubject);

Table altered.

SQL> desc Teachers;
Name                                     Null?    Type
-----
TEACHERID                                NOT NULL NUMBER(38)
TEACHERNAME                              VARCHA2(50)
TEACHINGSUBJECT                          VARCHA2(50)
```

-- Enforce constraint: The default value of Address is 'Chennai'

ALTER TABLE Student

MODIFY Address DEFAULT 'Chennai';

```
SQL> ALTER TABLE Student
2 MODIFY Address DEFAULT 'Chennai';

Table altered.

SQL> select*from Student;

ROLLNUMBER STUDENTNAME ADDRESS GENDER
-----
1 Raman Chennai
2 Payal Mumbai
3 Neha Delhi
4 Shruti Chennai
5 Ramesh Kolkata
```

-- Alter table to add Gender column with check constraint

ALTER TABLE Student

ADD Gender VARCHAR(10) CHECK (Gender IN ('Male', 'Female', 'Trans'));

```
SQL> ALTER TABLE Student ADD Gender VARCHAR(10) CHECK (Gender IN ('Male', 'Female', 'Trans'));
```

Table altered.

```
SQL> desc Student;  
Name
```

	Null?	Type
ROLLNUMBER		
STUDENTNAME	NOT NULL	NUMBER(38)
ADDRESS		VARCHAR2(50)
GENDER		VARCHAR2(100)
		VARCHAR2(10)

3.

-- Find the name of the Students who live in Chennai

```
SELECT StudentName
```

```
FROM Student
```

```
WHERE Address = 'Chennai';
```

```
SQL> SELECT StudentName  
2 FROM Student  
3 WHERE Address = 'Chennai';
```

```
STUDENTNAME
```

```
Raman  
Shruti
```

-- Find the name of the Teacher who teaches 'Cloud Computing' subject

```
SELECT TeacherName
```

```
FROM Teachers
```

```
WHERE TeachingSubject = 'Cloud Computing';
```

```
SQL> SELECT TeacherName  
2 FROM Teachers  
3 WHERE TeachingSubject = 'Cloud Computing';
```

```
TEACHERNAME
```

```
Sham Sharma
```

-- Find the Name of the teacher who teaches 'Java Programming' subject to the student 'Raman'

```
SELECT t.TeacherName
```

```
FROM Teachers t
```

```
JOIN College c ON t.TeacherID = c.TeacherID
```

```
JOIN Student s ON c.RollNumber = s.RollNumber
```

```
WHERE t.TeachingSubject = 'Java Programming' AND s.StudentName = 'Raman';
```

```
SQL> SELECT t.TeacherName  
2 FROM Teachers t  
3 JOIN College c ON t.TeacherID = c.TeacherID  
4 JOIN Student s ON c.RollNumber = s.RollNumber  
5 WHERE t.TeachingSubject = 'Java Programming' AND s.StudentName = 'Raman';
```

```
no rows selected
```

-- List out the RollNo and Name of students who have enrolled with the teacher Name 'Radha Krishnan'

```
SELECT s.RollNumber, s.StudentName
```

```
FROM Student s
```

```
JOIN College c ON s.RollNumber = c.RollNumber
```

```
JOIN Teachers t ON c.TeacherID = t.TeacherID
```

WHERE t.TeacherName = 'Radha Krishnan';

```
SQL> SELECT s.RollNumber, s.StudentName  
2 FROM Student s  
3 JOIN College c ON s.RollNumber = c.RollNumber  
4 JOIN Teachers t ON c.TeacherID = t.TeacherID  
5 WHERE t.TeacherName = 'Radha Krishnan';
```

ROLLNUMBER	STUDENTNAME
------------	-------------

5	Ramesh
---	--------

