

ASSIGNMENT 4

CASE STUDY:

Student is enrolled in a course. Each student is identified by his roll number. Other attributes of student are name, Dob, and semester. Each course has a unique course number. Other attributes of course are course name and number of students. Students are taught by faculty and faculties are working in course. Each faculty have a unique faculty number. Other attributes are name, course and salary. Draw an ERD and create the database for this scenario. Following constraint must be considered:

1. Student's roll number should be unique and can not be left blank.
2. Student name cannot be left blank.
3. Course number should be unique and cannot be left blank.
4. Any column of course table cannot be left blank.
5. Course can only be 'BCA', 'MCA', 'M.SC IT'
6. Faculty number should be unique and cannot be left blank.
7. Salary of faculty can not be a negative number.
8. You can assume other attributes of your choice.

1. Create Student table

QUERY:

```
1 v CREATE TABLE Student (  
2     Roll_Number INT PRIMARY KEY NOT NULL,  
3     Name VARCHAR(50) NOT NULL,  
4     DOB DATE,  
5     Semester INT  
6 );
```

Create Course Table

QUERY:

```
1 v CREATE TABLE Course (  
2     Course_Number INT PRIMARY KEY NOT NULL,  
3     Course_Name VARCHAR(50) NOT NULL,  
4     Number_of_Students INT NOT NULL,  
5     Course_Type VARCHAR(10) CHECK (Course_Type IN ('BCA', 'MCA', 'M.SC IT'))  
6 );
```

Create Faculty Table

QUERY:

```
1 v CREATE TABLE Faculty (  
2     Faculty_Number INT PRIMARY KEY NOT NULL,  
3     Name VARCHAR(50) NOT NULL,  
4     Course VARCHAR(50) NOT NULL,  
5     Salary DECIMAL(10,2) CHECK (Salary >= 0)  
6 );
```

Create Enroll table

QUERY:

```
1 v CREATE TABLE ENROLL (  
2     Roll_Number INT,  
3     FOREIGN KEY (Roll_Number) REFERENCES Student(Roll_NUMBER),  
4     Course_Number INT,  
5     FOREIGN KEY (Course_Number) REFERENCES Course(Course_number)  
6 );|
```

Create Works Table

QUERY:

```
1 v CREATE TABLE WORKS (  
2     Faculty_number INT,  
3     FOREIGN KEY (Faculty_number) REFERENCES Faculty(Faculty_number),  
4     Course_Number INT,  
5     FOREIGN KEY (Course_Number) REFERENCES Course(Course_number)  
6 );|
```

Create Teach table

QUERY:

```
1  
2 v CREATE TABLE Teach (  
3     Faculty_number INT,  
4     FOREIGN KEY (Faculty_number) REFERENCES Faculty(Faculty_number),  
5     Roll_Number INT,  
6     FOREIGN KEY (Roll_Number) REFERENCES Student(Roll_NUMBER)  
7 );|
```

Insert into Student table:

QUERY:

```
1  INSERT INTO STUDENT VALUES (0, 'S_0', '1-FEB-0', 0);
2  INSERT INTO STUDENT VALUES (1, 'S_1', '2-NOV-1', 1);
3  INSERT INTO STUDENT VALUES (2, 'S_2', '3-JAN-2', 2);
4  INSERT INTO STUDENT VALUES (3, 'S_3', '4-MAR-3', 3);
5  INSERT INTO STUDENT VALUES (4, 'S_4', '5-JAN-4', 4);
6  INSERT INTO STUDENT VALUES (5, 'S_5', '6-MAR-5', 5);
7  INSERT INTO STUDENT VALUES (11, 'S_11s', '11-NOV-11', 08);
```

1 row(s) inserted.

Insert into Course table:

```
1  INSERT INTO COURSE VALUES (200, 'C_200', 300, 'BCA');
2  INSERT INTO COURSE VALUES (201, 'C_201', 301, 'MCA');
3  INSERT INTO COURSE VALUES (202, 'C_202', 302, 'BCA');
4  INSERT INTO COURSE VALUES (203, 'C_203', 303, 'BCA');
5  INSERT INTO COURSE VALUES (204, 'C_204', 304, 'MCA');
```

1 row(s) inserted.

Insert into Faculty table:

```
1  INSERT INTO FACULTY VALUES (100, 'F_100', 'BCA', 10000);
2  INSERT INTO FACULTY VALUES (101, 'F_101', 'BCA', 10100);
3  INSERT INTO FACULTY VALUES (102, 'F_102', 'M.SC IT', 10200);
4  INSERT INTO FACULTY VALUES (103, 'F_103', 'BCA', 10300);
5  INSERT INTO FACULTY VALUES (104, 'F_104', 'M.SC IT', 10400);
```

1 row(s) inserted.

INSERT into Enroll, Teach, Works tables:

```
1  INSERT INTO ENROLL VALUES (1, 200);
2  INSERT INTO ENROLL VALUES (2, 200);
3  INSERT INTO ENROLL VALUES (3, 204);
4  INSERT INTO ENROLL VALUES (2, 202);
5  INSERT INTO ENROLL VALUES (4, 202);
6
7  INSERT INTO WORKS VALUES (101, 202);
8  INSERT INTO WORKS VALUES (102, 204);
9  INSERT INTO WORKS VALUES (103, 201);
10 INSERT INTO WORKS VALUES (104, 202);
11
12 INSERT INTO TEACH VALUES (101, 2);
13 INSERT INTO TEACH VALUES (101, 1);
14 INSERT INTO TEACH VALUES (103, 4);
15 INSERT INTO TEACH VALUES (102, 1);
```

1 row(s) inserted.

1. Retrieve the information of those students who are taught by faculty 101.

QUERY:

```
1 v SELECT * FROM Student WHERE Roll_Number IN (  
2     SELECT Roll_number FROM Teach WHERE Faculty_Number = 101  
3 );
```

ROLL_NUMBER	NAME	DOB	SEMESTER
1	S_1	02-NOV-01	1
2	S_2	03-JAN-02	2

2. Retrieve record of faculty whose faculty number = 104.

QUERY:

```
1 SELECT * FROM Faculty WHERE faculty_number = 104;
```

FACULTY_NUMBER	NAME	COURSE	SALARY
104	F_104	M.SC IT	10400

3. Display the information of the students whose name ends with 's'.

QUERY:

```
1 SELECT * FROM Student WHERE Name LIKE '%s';|
```

ROLL_NUMBER	NAME	DOB	SEMESTER
11	S_11s	11-NOV-11	8