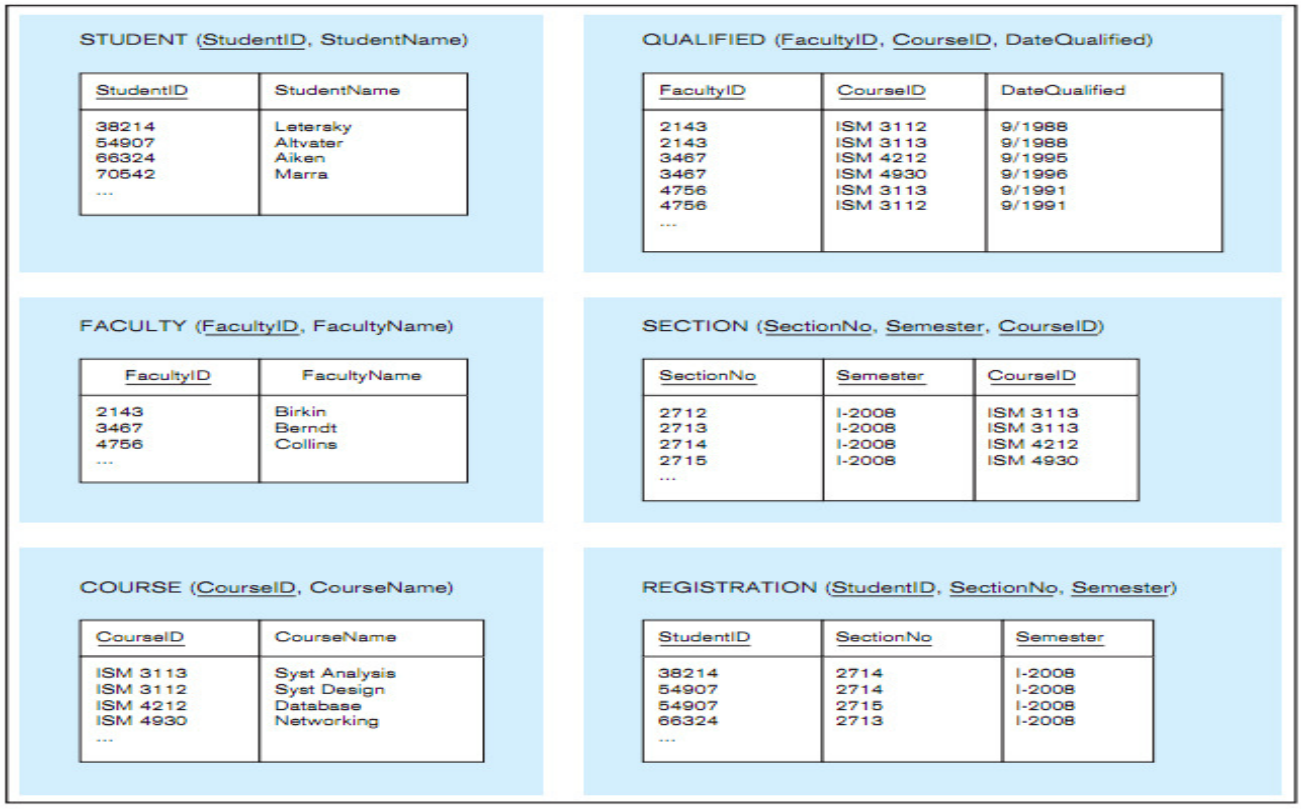
TUTORIAL 6

1. Write a database description for each of the relations shown, using SQL DDL (shorten, abbreviate, or change any data names, as needed for your SQL version). Assume the following attribute data types:

StudentID (integer, primary key), StudentName (25 characters), FacultyID (integer, primary key), FacultyName (25 characters), CourseID (8 characters, primary key), CourseName (15 characters), DateQualified (date), SectionNo (integer, primary key), Semester (7 characters)



STUDENT (StudentID, StudentName)

StudentID	StudentName
38214	Letersky
54907	Altvater
66324	Aiken
70542	Marra
...	

QUALIFIED (FacultyID, CourseID, DateQualified)

FacultyID	CourseID	DateQualified
2143	ISM 3112	9/1988
2143	ISM 3113	9/1988
3467	ISM 4212	9/1995
3467	ISM 4930	9/1996
4756	ISM 3113	9/1991
4756	ISM 3112	9/1991
...		

FACULTY (FacultyID, FacultyName)

FacultyID	FacultyName
2143	Birkin
3467	Berndt
4756	Collins
...	

SECTION (SectionNo, Semester, CourseID)

SectionNo	Semester	CourseID
2712	I-2008	ISM 3113
2713	I-2008	ISM 3113
2714	I-2008	ISM 4212
2715	I-2008	ISM 4930
...		

COURSE (CourseID, CourseName)

CourseID	CourseName
ISM 3113	Syst Analysis
ISM 3112	Syst Design
ISM 4212	Database
ISM 4930	Networking
...	

REGISTRATION (StudentID, SectionNo, Semester)

StudentID	SectionNo	Semester
38214	2714	I-2008
54907	2714	I-2008
54907	2715	I-2008
66324	2713	I-2008
...		

2. Create an SQL VIEW for foollowing table

StudentID	StudentName
38214	Letersky
54907	Altvater
54907	Altvater

3. Write SQL data definition commands for each of the following queries:

a. How would you add an attribute, Class, to the Student table?

b. How would you remove the Registration table?

c. How would you change the FacultyName field from 25 characters to 40 characters?

4. Write SQL commands for the following:

a. Create two different forms of the INSERT command to add a student with a student ID of 65798 and last name Lopez to the Student table.

b. Now write a command that will remove Lopez from the Student table.

c. Create an SQL command that will modify the name of course ISM 4212 from Database to Introduction to Relational Databases.

5. Write SQL queries to answer the following questions:

a. Which students have an ID number that is less than 50000?

b. What is the name of the faculty member whose ID is 4756?

c. What is the smallest section number used in the first semester of 2008?

6. Write SQL queries to answer the following questions:

a. How many students are enrolled in Section 2714 in the first semester of 2008?

b. Which faculty members have qualified to teach a course since 1993? List the faculty ID, course, and date of qualification.

7. Write SQL queries to answer the following questions:

a. Which students are enrolled in Database and Networking? (Hint: Use SectionNo for each class so you can determine the answer from the Registration table by itself.)

b. Which instructors cannot teach both Syst Analysis and Syst Design?

8. Write SQL queries to answer the following questions:

a. What are the courses included in the Section table? List each course only once.

b. List all students in alphabetical order by StudentName.

c. List the students who are enrolled in each course in Semester I, 2008. Group the students by the sections in which they are enrolled.

d. List the courses available. Group them by course prefix. (ISM is the only prefix shown, but there are many others throughout the university.)

SOLUTION

1.

```
CREATE TABLE STUDENT (
STUDENT_ID INTEGER UNSIGNED NOT NULL,
STUDENT_NAME VARCHAR(25),
CONSTRAINT STUDENT_PK PRIMARY KEY (STUDENT_ID)
);

CREATE TABLE FACULTY
(
FACULTY_ID INTEGER UNSIGNED NOT NULL,
FACULTY_NAME VARCHAR(25),
CONSTRAINT FACULTY_PK PRIMARY KEY (FACULTY_ID)
);

CREATE TABLE COURSE

        (COURSE_ID          CHAR(8)          NOT NULL,

        COURSE_NAME         VARCHAR(15),

        CONSTRAINT COURSE_PK PRIMARY KEY (COURSE_ID));

CREATE TABLE SECTION
(
        SECTION_NO  INTEGER                  UNSIGNED NOT NULL,

        SEMESTER    CHAR(7)                  NOT NULL,

        COURSE_ID                CHAR(8),
CONSTRAINT SECTION_PK PRIMARY KEY (SECTION_NO, SEMESTER),
CONSTRAINT SECTION_FK FOREIGN KEY (COURSE_ID)
REFERENCES COURSE (COURSE_ID)
);

CREATE TABLE QUALIFIED

        (FACULTY_ID  INTEGER                 UNSIGNED NOT NULL ,

        COURSE_ID    CHAR(8)                 NOT NULL,

        DATE_QUALIFIED      DATE,
```

CONSTRAINT QUALIFIED_PK PRIMARY KEY (FACULTY_ID, COURSE_

CONSTRAINT QUALIFIED_FACULTY_FK FOREIGN KEY
(FACULTY_ID) REFERENCES FACULTY (FACULTY_ID),

CONSTRAINT QUALIFIED_COURSE_FK FOREIGN KEY (COURSE_ID)
REFERENCES COURSE (COURSE_ID));

CREATE TABLE REGISTRATION

(STUDENT_ID INTEGER UNSIGNED NOT NULL,

SECTION_NO INTEGER UNSIGNED NOT NULL,

SEMESTER CHAR(7) NOT NULL,

CONSTRAINT REGISTERED_PK PRIMARY KEY (STUDENT_ID,
SECTION_NO, SEMESTER),

CONSTRAINT STUDENT_REGISTERED_FK FOREIGN KEY(STUDENT_ID)
REFERENCES STUDENT(STUDENT_ID),

CONSTRAINT COURSE_REGISTERED_FK FOREIGN KEY (SECTION_NO,
SEMESTER) REFERENCES SECTION(SECTION_NO, SEMESTER));

2. CREATE VIEW STUDENT_VIEW AS SELECT STUDENT_ID, STUDENT_NAME FROM STUDENT;

3.

a. ALTER TABLE STUDENT ADD CLASS VARCHAR(5)

b. DROP TABLE REGISTRATION;

c. ALTER TABLE FACULTY MODIFY FACULTY_NAME VARCHAR(50);

4.

a. INSERT INTO STUDENT VALUES (12345, "LOPEZ");

INSERT INTO STUDENT (STUDENT_ID, STUDENT_LNAME) VALUES (12345, "Kloser");

b. DELETE FROM STUDENT WHERE STUDENT_NAME = 'LOPEZ';

C. UPDATE COURSE SET COURSE_NAME = 'INTRODUCTION TO RELATIONAL DATABASE' WHERE COURSE_ID= 'ISM 4212';

5.

A. SELECT STUDENT_ID, STUDENT_NAME FROM STUDENT WHERE STUDENT_ID <50000;

B. SELECT STUDENT_NAME FROM STUDENT WHERE STUDENT_ID = 4756;

C. SELECT MIN(SECTION_NO) FROM REGISTRATION WHERE SEMESTER= 'I-2008';

6.

A. SELECT COUNT(STUDENT_ID) FROM REGISTRATION WHERE SECTION_NO =2714 AND SEMESTER= 'I-2008'

B. SELECT Q.FACULTY_ID, C.COURSE_NAME, Q.DATE_QUALIFIED

FROM FACULTY AS F, COURSE AS C, QUALIFIED AS Q

WHERE YEAR(Q.DATE_QUALIFIED) >= 1993

AND C.COURSE_ID =Q.COURSE_ID;

```
SELECT FACULTY_ID, COURSE_ID, DATE_QUALIFIED
        FROM QUALIFIED
        WHERE DATE_QUALIFIED >= '1993-01-01';
```

7. a,

```
SELECT STUDENT_ID, COUNT(*)
        FROM REGISTRATION
        WHERE SECTION_NO IN (2714,2715)
        GROUP BY STUDENT_ID
        HAVING COUNT(*) > 1;
```

b. SELECT FACULTY_ID,COUNT(*)FROM QUALIFIED WHERE

```
        COURSE_ID IN ('ISM 3113','ISM 3112')
        GROUP BY INSTRUCTOR_ID
        HAVING COUNT(*) = 1;
```

8

A. SELECT DISTINCT COURSE_ID FROM SECTION;

B. B. SELECT STUDENT_NAME

```
        FROM STUDENT
```

```
        ORDER BY STUDENT_NAME;
```

C.
```
SELECT SECTION_NO,SEMESTER,STUDENT_ID
        FROM REGISTRATION
        WHERE SEMESTER = 'I-2008'
        ORDER BY SECTION_NO, SEMESTER, STUDENT_ID;
```

D.
```
SELECT COURSE_ID,COURSE_NAME
        FROM COURSE
        ORDER BY COURSE_ID;
```