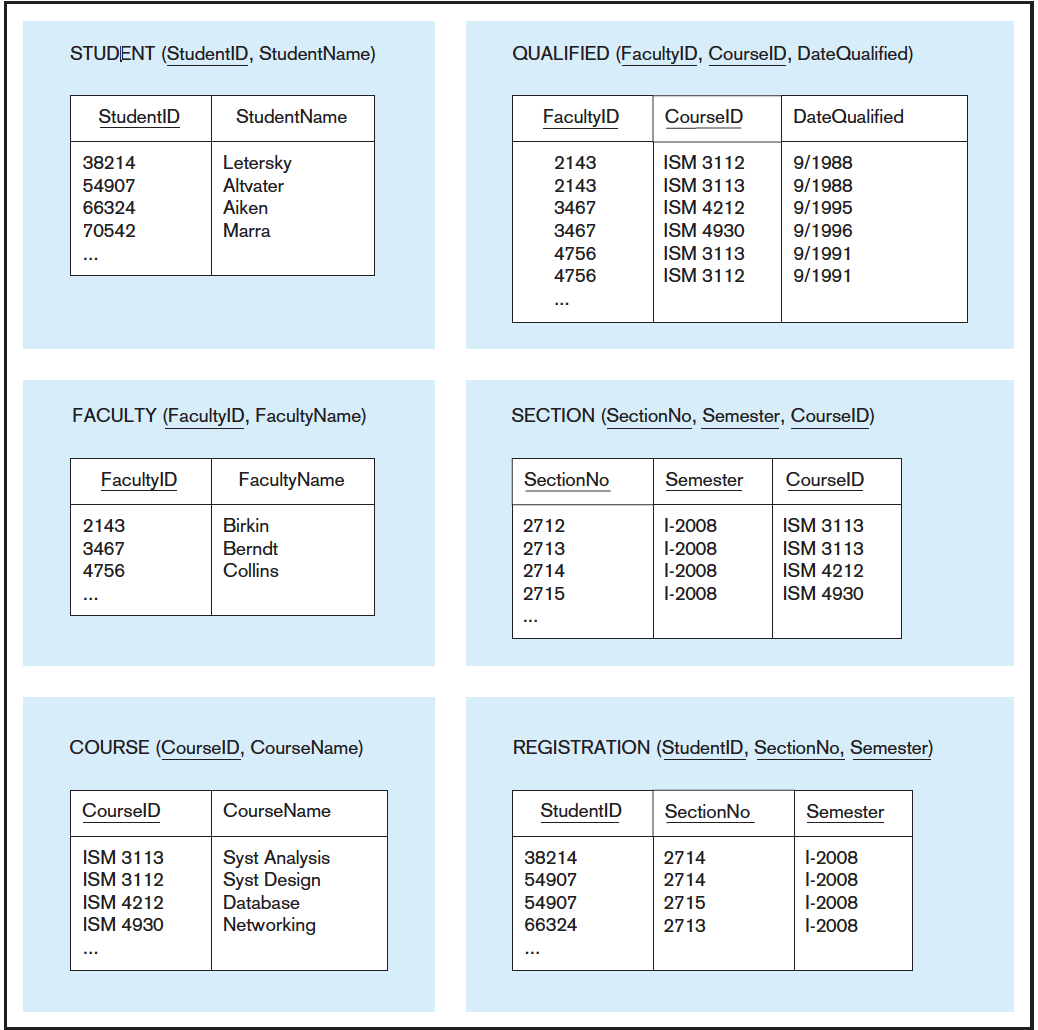
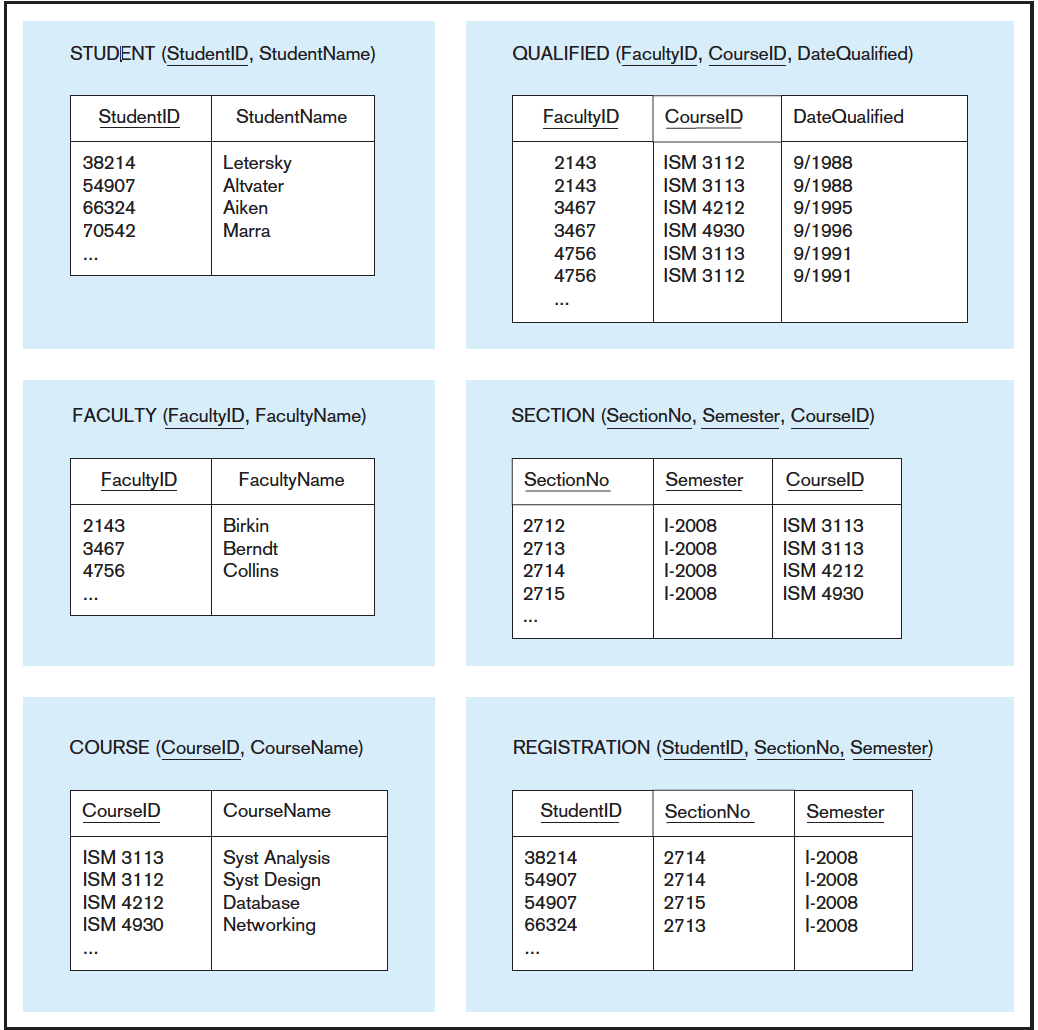
Tutorial 6

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:

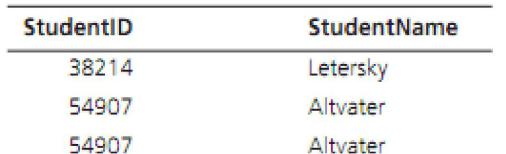
StudentID (integer, primary key), StudentName (25 characters), FacultyID (integer, primary key), FacultyName (25 characters), CourseID (8 characters, primary key), CourseName (15 characters), DateQualified (date), SectionNo (integer, primary key), Semester (7 characters)





FROM Question >=2, use table above to write SQL Query, Faculty has the same meaning as Instructor.

2. Create an SQL VIEW for following table



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

a. How would you add an attribute: Class - Varchar(5), 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?

1. 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 only.

7. 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 information in alphabetical order by StudentName.

**SOLUTION**

**1.**

CREATE TABLE STUDENT

(

STUDENTID INTEGER UNSIGNED NOT NULL PRIMARY KEY,

STUDENTNAME VARCHAR(25)

);

CREATE TABLE FACULTY

(

FACULTYID INTEGER UNSIGNED NOT NULL PRIMARY KEY,

FACULTYNAME VARCHAR(25),

);

CREATE TABLE COURSE

(

COURSEID CHAR(8) NOT NULL PRIMARY KEY,

COURSENAME VARCHAR(15),

)

CREATE TABLE SECTION

(

SECTIONNO

INTEGER

UNSIGNED NOT NULL,

SEMESTER

CHAR(7)

NOT NULL,

COURSEID

CHAR(8),

PRIMARY KEY (SECTION\_NO, SEMESTER),

FOREIGN KEY (COURSE\_ID) REFERENCES COURSE (COURSE\_ID)

);

CREATE TABLE QUALIFIED

(FACULTYID

INTEGER

UNSIGNED NOT NULL,

COURSEID

CHAR(8)

NOT NULL,

DATEQUALIFIED DATE,

PRIMARY KEY (FACULTY\_ID, COURSEID),

FOREIGN KEY (FACULTY\_ID) REFERENCES FACULTY (FACULTY\_ID),

FOREIGN KEY (COURSE\_ID) REFERENCES COURSE (COURSE\_ID)

);

|  |  |  |
| --- | --- | --- |
| CREATE TABLE REGISTRATION |  |  |
| (STUDENTID | INTEGER | UNSIGNED NOT NULL, |
| SECTIONNO | INTEGER | UNSIGNED NOT NULL, |
| SEMESTER | CHAR(7) | NOT NULL, |

PRIMARY KEY (STUDENT\_ID, SECTION\_NO, SEMESTER),

FOREIGN KEY(STUDENT\_ID) REFERENCES STUDENT(STUDENT\_ID),

FOREIGN KEY (SECTION\_NO) REFERENCES SECTION(SECTION\_NO);

1. CREATE VIEW STUDENT\_VIEW AS SELECT STUDENTID, STUDENTNAME FROM STUDENT;

3.

a. ALTER TABLE STUDENT ADD CLASS VARCHAR(5)

b. DROP TABLE REGISTRATION;

1. ALTER TABLE FACULTY MODIFY FACULTY\_NAME VARCHAR(50);

4.

a. INSERT INTO STUDENT VALUES (65798, “LOPEZ”);

INSERT INTO STUDENT (STUDENTID, STUDENTNAME) VALUES (65798, “LOPEZ”);

b. DELETE FROM STUDENT WHERE STUDENTNAME = ‘LOPEZ’;

1. UPDATE COURSE SET COURSENAME = ‘INTRODUCTION TO RELATIONAL DATABASE’ WHERE COURSEID= ‘ISM 4212’;

5.

1. SELECT STUDENTID, STUDENTNAME FROM STUDENT WHERE STUDENTID <50000;
2. SELECT STUDENTNAME FROM STUDENT WHERE STUDENTID = 4756;
3. SELECT MIN(SECTIONNO) FROM REGISTRATION WHERE SEMESTER= ‘I-2008’;

6.

1. SELECT COUNT(STUDENTID) FROM REGISTRATION WHERE SECTION\_NO =2714 AND SEMESTER= ‘I-2008’

B. SELECT FacultyID WHERE YEAR(DATEQUALIFIED) >= 1993

SELECT FACULTY\_ID FROM QUALIFIED WHERE DATEQUALIFIED >= ‘1993-01-01';

7. a. SELECT DISTINCT(CourseID) FROM Section

SELECT CourseID FROM Section GROUP BY CourseID

b. SELECT \* FROM Student ORDER BY StudentName