Readings:

* Multiple Table Processing: JOINS
  + chapter 7: “Advanced SQL” (Hoffer, Ramesh, & Topi) page 289- 298
  + chapter 6: “Queries: JOIN Operator” (Petkovic) from page 240 -255
* Multiple Table Processing: Subquery
  + chapter 6: “Queries: Correlated Subqueries” (Petkovic) from page 255 to the end of the chapter
  + chapter 7: “Advanced SQL” (Hoffer, Ramesh, & Topi) from page 289 -310

**Part A**

* + Chapter 7(Hoffer, Ramesh, & Topi)
  + Problems and Exercises 1,2,3,4,5

1.

1. Display the course ID and course name for all courses with an ISM prefix:

**Query:**  
SELECT CourseID, CourseName

FROM Course

WHERE CourseID LIKE ‘ISM%’;

1. Display all courses for which Professor Berndt has been qualified:

**Query:**  
SELECT f.FacultyName, c.CourseID,c.CourseName

FROM Faculty f JOIN Qualified q ON f.FacultyID = q.FacultyID

JOIN Course c ON c.CourseID=q.CourseID;

WHERE q.FacultyID = 3467

1. Display the class roster, including student name, for all students enrolled in section 2714 of ISM 4212:

**Query:**  
SELECT s.StudentID, s.StudentName,

sc.CourseID, r.SectionNo, Semester

FROM Section sc JOIN Registration r ON sc.SectionNo= r.SectionNo

JOIN Student s ON s.StudentID= r.StudentID

WHERE

r.SectionNo=2714

AND sc.Semester=‘I-2008’

AND r.Semester=‘I-2008’

ORDER BY s.StudentName;

2. Which instructors are qualified to teach ISM 3113?

**Query:**

SELECT Faculty.FacultyName

FROM Faculty f JOIN Qualified q ON f.FacultyID = q.FacultyID

WHERE q.CourseID=‘ISM 3113’;

3. Is any instructor qualified to teach ISM 3113 and not qualified to teach ISM 4930?

**Query:**

SELECT Faculty.FacultyName

FROM Faculty f JOIN Qualified q ON f.FacultyID = q.FacultyID

WHERE

q.CourseID=‘ISM 3113’

AND NOT (q.CourseID=‘ISM 4930’);

## 4*.*

a. How many students are enrolled in section 2714 during semester I-2008?

**Query:**  
SELECT COUNT (DISTINCT (StudentID))

FROM Registration

WHERE SectionID = 2714

AND Semester = ‘I-2008’;

1. How many students are enrolled in ISM 3113 during semester I-2008?

**Query:**  
SELECT COUNT (DISTINCT (r.StudentID))

FROM Section s JOIN Registration r ON s.SectionNo = r.SectionNo

WHERE s.CourseID = ‘ISM 3113’

AND r.Semester = ‘I-2008’;

5. Which students were not enrolled in any courses during semester I-2008?

**Query:**

SELECT DISTINCT StudentID, Student\_NAME

FROM Student s

WHERE NOT EXISTS

(SELECT \* FROM Registration

WHERE s.StudentID =

r.StudentID

AND Semester= ‘I-2008’);

Part B

Use the “sample\_DDL\_script.sql” file from the blackboard to create a sample database named “sample”. Use a second script file named “sample\_insert\_script.sql” to populate the database with sample data. Both files can be downloaded from the class blackboard.

Using the database created, answer write the SQL statements for the following questions:

1. Return a list with dept\_no and a count of employees in each department.

select dept\_no, count(\*)

from [dbo].[employee]

group by dept\_no

1. Return a list with dept\_no and a count of employees in each department. Only show departments with greater than or equal to 2 employees

select dept\_no, count(\*)

from [dbo].[employee]

group by dept\_no

having count(\*)>1

1. Find the highest employee number.

select max(emp\_no) from employee

1. What is the difference between COUNT(\*) and COUNT(column)? Write a SQL Example

Count(\*) counts every record in the table because every column is being considered. Count(column) only counts records without NULL values int that respective column

Examples:

select count(\*) from [dbo].[works\_on]

select count(job) from [dbo].[works\_on]

1. Get the project\_no that are done by more than 1 employee.

select project\_no, count(emp\_no)

from [dbo].[works\_on]

group by project\_no

having count(emp\_no)>1