**Introduction To Database Management Systems (Lab 2)**

**Problem:**

Problem Part 1.

Create a table for SNU courses

1. courseID (varchar, size 6, primary key)

2. courseName (varchar, size 50, not null)

3. courseDept (varchar, size 25, not null)

4. courseSchool(varchar, size 50, not null)

Problem Part 2.

Create a table for SNU faculty

1. courseID (varchar, size 6, foreign key)

2. facultyName (varchar, size 30, not null)

3. facultyPosition (varchar, size 30, not null)

4. facultyQualification (varchar, size 30, not null, default ’unknown’)

Problem Part 3.

Create a table for SNU students

1. courseID (varchar, size 6, foreign key)

2. studentName (varchar, size 30, not null)

3. studentYear(integer, size 1, not null)

4. studentMajor (varchar, size 30, not null)

5. studentGrade (varchar, size 1, not null, default ’I’)

Problem Part 4.

Insert Data into the tables as per Department of Computer Science,

School of Engineering. (At least 8 records for courses + 2 faculty teaching more than one

courses + At least 10 records for students + 3 students enrolled in more than one course +

Leave one course without students.)

Problem Part 5.

Execute these queries on the database

(a). PROJECT only courseID and courseName from the courses table.

(b). PROJECT only studentName and courseID where studentGrade is Incomplete (I).

(c). PROJECT only facultyName (RENAME as ’Name’ ) and facultyQualification (RE-

NAME as ’Qualification’) for faculty where courseName starts with ’D’ (e.g. Data

Structures and Database Systems).

(d). SELECT all info of students who are either in year 1 or in year 4 (through UNION

operator).

(e). PROJECT studentName of students who are enrolled in both Database Systems and

Operating Systems (through INTERSECTION operator).

(f). The CARTESIAN JOIN or CROSS JOIN returns the Cartesian product of the sets of

records from the two or more joined tables. Thus, it equates to an inner join where

the join-condition always evaluates to True or where the join-condition is absent from

the statement. Create cartesian product table of courses and students.

(g). PROJECT studentName,studentYear,studentMajor of those whose courseID is IN ’Data

Structures’,’Operating Systems’

(h). SELECT all info from faculty where facultyName starts with ’D’.

(i). SELECT studentName (RENAME as ’Name’) from students where studentYear is

between 2 and 4 inclusive.