**Assignment 1: Basic SQL**

1. Write the following simple SQL Queries on the University Schema
   1. Find the names of all the students whose total credits are greater than 100
   2. Find the course id and grades of all courses taken by any student named 'Tanaka'
   3. Find the ID and name of instructors who have taught a course in the Comp. Sci. department, even if they are themselves not from the Comp. Sci. department. To test this query, make sure you add appropriate data, and include the corresponding insert statements along with your query.
   4. Find the courses which are offered in both 'Fall' and 'Spring' semester (not necessarily in the same year).
      1. Find the names of all the instructors from Comp. Sci. department
      2. Find the course id and titles of all courses taught by an instructor named 'Srinivasan'
      3. Find names of instructors who have taught at least one course in Spring 2009

Aggregate Function

1. Find the number of instructors who have never taught any course. If the result of your query is empty, add appropriate data (and include corresponding insert statements) to ensure the result is not empty.
2. Find the total capacity of every building in the university
3. Find the maximum number of teachers for any single course section.  Your output should be a single number.  For example if CS-101 section 1 in Spring 2012 had 3 instructors teaching the course, and no other section had more instructors teaching the section, your answer would be 3.
4. Find all departments that have at least one instructor, and list the names of the departments along with the number of instructors;   order the result in descending order of number of instructors.
5. As in the previous question, but this time you shouold include departments even if they do not have any instructor, with the count as 0
6. For each student, compute the total credits they have successfully completed, i.e. total credits of courses they have taken, for which they have a non-null grade other than 'F'. Do NOT use the tot\_creds attribute of student.
7. Find the number of students who have been taught (at any time) by an instructor named 'Srinivasan'. Make sure you count a student only once even if the student has taken more than one course from Srinivasan.
8. Find the name of all instructors who get the highest salary in their department.
9. Find all students who have taken all courses taken by instructor 'Srinivasan'. (This is the division operation of relational algebra.) You can implement it by counting the number of courses taught by Srinivasan, and for each student (i.e. group by student), find the number of courses taken by that student, which were taught by Srinivasan. Make sure to count each course ID only once.
10. Find the total money spent by each department for salaries of instructors of that department.
11. Find the names of all students whose advisor has taught the maximum number of courses (multiple offerings of a course count as only 1).

Nested Queries

1. Find the id and title of all courses which do not require any prerequisites.
2. Find the names of students who have not taken any biology dept courses
3. Write SQL update queries to perform the following (queries 2 and 4 are pretty meangless, but still fun to write):
   * 1. Give a 10% hike to all instructors
     2. Increase the tot\_creds of all students who have taken the course titled "Genetics" by the number of credits associated with that course.
     3. For all instructors who are advisors of atleast 2 students, increase their salary by 50000.
     4. Set the  credits to 2 for all courses which have less than 5 students taking them (across all sections for the course, across all years/semesters).