

## Homework 2

assigned October 16; due October 30 5:00pm

For this assignment, you create a database, populate it with data, and program queries to be run against the database. This part must be done on db2.

You can work in groups of up to 3 people. When you submit the assignment, do not forget to indicate the section you are in. Groups of students from different sections are **not allowed**. If such an assignment is submitted, it will be arbitrarily assigned to one section, and student(s) from the other section will get no marks for it.

You must submit electronically, and provide a printout as well (one submission per group). The electronic submission must the following files (with these exact names):

- group.txt — text file with names and student numbers of all members of the group;
- a2.ddl — all DDL statements;
- a2.insert — all insert statements;
- a2.drop — drop statements for all databases and views that you create;
- a2.sql — your SQL queries

### Database description

Consider a database with the tables listed below.

```
person[sin,name,gender,dob] (dob = date of birth)
address[sin,number,street,city,phone]
doctor[name,department,hospital,city,salary]
patient[sin,disease,bed,room,hospital,fee,entry_date,exit_date]
treat[doctor_name,patient_name]
```

The followings are assumed to be true:

- each person has a unique addresses and a unique SIN number;
- no two names are the same;
- every doctor and patient is a person;
- the gender of a patient is either 'male' or 'female';
- there are no duplicates in the treat table.

**Part I (30 points)**

Write DDL statements for these tables, and populate them with data. You should invent some reasonably looking data. It is required that each table contains at least 5 tuples, although you may find it convenient to have larger tables for debugging.

## Part II

Write the following queries. Each is worth **5 points**.

1. Find names and addresses of all people who live in Toronto.
2. Find names and salaries of all doctors who are currently treating ulcers, and who earn less than \$100,000.
3. Find name, age and phone number of every patient whose fee is higher than \$30,000.
4. Find the SIN of every doctor who has at least one male patient older than 75.
5. Find the name of every doctor who lives and works in the same city.
6. Use SQL to check if the following is true: each doctor should be affiliated with the same hospital where all his/her patients are. Your output could be a list of doctors who violate this rule.
7. Change the city of John Smith from Toronto to Oshawa, and change his phone number to (905)1111111.
8. Provide a 3% discount to all patients in a room that has more than 2 patients.
9. For every doctor, find his/her SIN and the SIN of every patient treated by the doctor.
10. Use SQL to check if the following is true: there are no patients who are being treated for 5 or more different diseases at the same time. Again, your output should describe violations of this condition.

**Part III.** Write the following in SQL. Each query is worth **10 points**.

11. Find the name, room and hospital of every patient who is treated by more than one doctor.
12. Find the department(s) at the Mount Sinai hospital in Toronto that have the fewest number of doctors.
13. Find names of patients who pay more than the average fee of all patients in the same hospital.
14. For each doctor, retrieve the doctor's name and department and those rooms that have at least two of the doctor's patients.
15. The annual revenue of each hospital is defined as the sum of all fees paid by the patients who were discharged in that year. List top 10 hospitals in terms of their annual revenue in 2002.
16. Check if the following is true: there are no male and female patients in the same room. This time, return a table with a single entry which is either "true" or "false".
17. Check if the following is true: there was a day when, at some hospital, more than 30% of all the patients were being treated for the same disease. Again, the output should be a table with a single entry which is either "true" or "false" (that is, you do *not* have to output dates and hospitals).

**TOTAL: 150 points.**