University of Toronto
CSC343, Fall 2016

Assignment 1

Due: Wednesday 12 October at 8:00 pm sharp!

Learning Goals

By the end of this assignment you should have:

1. cleared up any gaps in your understanding of the relational model,
2. become comfortable reading and understanding a new schema,
3. mastered the individual techniques for writing relational algebra queries and integrity constraints that we learned in class,
4. practised applying them to complex problems, and
5. become well prepared to learn SQL.

Later in the course, you will learn about how to develop your own schema based on knowledge of the domain. Even though developing a schema is really the first step in building a database, it is a more advanced task than querying an existing database; this is why we will be learning about it and practising it later.

Schema

For this assignment, you will operate on the database for an Ontario pharmacy. Here is its schema.

Relations

- Product(DIN, manufacturer, name, form, schedule)
  A tuple in this relation represents a drug product that can be purchased from the pharmacy. *DIN* is the Drug Identification Number, an eight-digit number assigned to each drug product authorized for use in Canada, *manufacturer* is the name of the manufacturer, *name* is the name of the drug, for example, "ADALAT XL - SRT 30MG", *form* is the form in which the drug product is produced, such as "capsule", and *schedule* is the category in which the federal government places the drug, for example "narcotic".
- ActiveIngredient(DIN, name, strength, unit)
  A tuple in this relation represents an active ingredient in a drug product. *DIN* is the Drug Identification Number, *name* is the name of the active ingredient, *strength* is the strength of the active ingredient, such as 200, and *unit* is the units in terms of which the strength is expressed, such as "mg".
- Administration(DIN, route)
  A tuple in this relation represents the route of administration of a drug product, that is, how it is administered. *DIN* is the Drug Identification Number, and *route* is the route of administration, such as "oral" or "intravenous".
- Price(DIN, price)
  A tuple in this relation represents the price of a drug product *DIN* is the Drug Identification Number, and *price* is its price.

- Prescription(RxID, date, patient, drug, doctor, dosage, note)
  A tuple in this relation represents a prescription. *RxID* is the prescription ID, *date* is the date on which it was written, *patient* is the OHIP number of the patient it was written for, *drug* is the drug product it is a prescription for, *doctor* is the identification number of the doctor who wrote it, and *dosage* is the dosage of the prescription.

- Pharmacist(OCP, name, registered)
  A tuple in this relation represents the fact that an pharmacist is registered with the Ontario College of Pharmacists. *OCP* is their Ontario College of Pharmacists identification number, *name* is their name, and *registered* is the date on which they were registered.

- Filled(RxID, date, pharmacist)
  A tuple in this relation represents the fact that a prescription was filled. *RxID* is the prescription ID, *date* is the date on which it was filled, and *pharmacist* is the pharmacist who filled the prescription.

- TrainedUnder(P1, P2, completed)
  A tuple in this relation represents the fact that P1 completed a training internship under the supervision of P2. We say that "P1 trained under P2", or the "P2 trained P1". *P1* and *P2* are pharmacist's OCP numbers, and *completed* is the date on which the training was completed.

Integrity constraints

- ActiveIngredient[DIN] $\subseteq$ Product[DIN]
- Administration[DIN] $\subseteq$ Product[DIN]
- Price[DIN] $\subseteq$ Product[DIN]
- Prescription[drug] $\subseteq$ Product[DIN]
- Filled[RxID] $\subseteq$ Prescription[RxID]
- Filled[pharmacist] $\subseteq$ Pharmacist[OCP]
- TrainedUnder[P1] $\subseteq$ Pharmacist[OCP]
- TrainedUnder[P2] $\subseteq$ Pharmacist[OCP]
- Π_{schedule}Product $\subseteq$ {"prescription", "narcotic", "OTC"}

Part 1: Queries

Write the queries below in relational algebra. There are a number of variations on relational algebra, and different notations for the operations. You must use the same notation as we have used in class and on the slides. You may use assignment, and the operators we have used in class: $\Pi, \sigma, \bowtie, \bowtie_{condition}, \times, \cap, \cup, -, \rho$. Assume that all relations are sets (not bags), as we have done in class, and do not use any of the extended relational algebra operations from Chapter 5 of the textbook (for example, do not use the extended projection).

Some additional points to keep in mind:

- Do not make any assumptions about the data that are not enforced by the original constraints given above, including the ones written in English. Your queries should work for any database that satisfies those constraints.

- Assume that every tuple has a value for every attribute. For those of you who know some SQL, in other words, there are no null values.
- Remember that the condition on a select operation may only examine the values of the attributes in one tuple, not whole columns. In other words, to use a value (other than a literal value such as 100 or "Adele"), you must get that value into the tuples that your select will examine.
- The condition on a select operation can use comparison operators (such as $\leq$ and $\neq$) and boolean operators ($\vee$, $\wedge$ and $\neg$). Simple arithmetic is also okay, *e.g.*, attribute1 $\leq$ attribute2 + 5000.
- Two relations in our schema have a date attribute. You may use comparison operators on such values. You may refer to the year component of a date and time attribute d using the notation d.year.
- You are encouraged to use assignment to define intermediate results.
- It's a good idea to add commentary explaining what you're doing. This way, even if your final answer is not completely correct, you may receive part marks.
- The order of the columns in the result doesn't matter.
- When asked for a maximum or minimum, if there are ties, report all of them.

At least one of the queries cannot be expressed in the language that you are using. In those cases, simply write "cannot be expressed". Note: The queries are not in order according to difficulty.

1. Find all patients who (a) have had more than 2 different doctors write them a prescription, and (b) have had a narcotic prescribed to them by every doctor who has written them a prescription. A narcotic is a drug whose schedule is "narcotics". Report the patient's OHIP number.
2. Find every prescription from 2016 that has never been filled. Report the patient's OHIP number, the prescription ID, prescription date, and drug.
3. Find the pharmacist who has trained the most people. Report the pharmacist's OCP number and name.
4. The "narcotics prescription period" of a doctor for a patient is the time from the first prescription for narcotics from that doctor for that patient to the most recent one. (It would be zero if that doctor wrote only one prescription for narcotics for that patient.) Find all patients who have had narcotics prescribed by two or more doctors, and for whom the narcotics prescription periods never overlap. In other words, if they had narcotics prescribed by n different doctors,

   $$[start_1..end_1] < [start_2..end_2] < ... < [start_n..end_n]$$

   where $start_i$ and end_i are the start and end of the narcotics prescription period of doctor i for that patient. Notice that we have written strictly less than. This means that if $end_i = start_{i+1}$ we do not consider that the periods overlap. Report the patient's OHIP number.
5. Find all pharmacists who have never filled a prescription for a drug product whose active ingredient is "codeine". Report their OCP number and every schedule for which they *have* filled a prescription. Put the information into a relation with attributes "OCP" and "schedule".
6. Lets say a minor trainer is a pharmacist who has trained no more than two people. (They may have trained none.) Find all pharmacists who have trained 2 or more minor trainors. (They may have trained other pharmacists who were not minor trainors.) Report the pharmacist's OCP number.

7. Find the most junior pharmacist: the pharmacist whose first time filling a prescription has the latest date. Report the pharmacist's OCP number, the prescription ID for the first prescription they filled, the date on which it was written, and the date on which it was filled.

8. Find every patient who has had a prescription for a homeopathic drug product filled, that is, a product whose schedule is "homeopathic", but has never had a prescription filled for a drug product with any other schedule.

9. Find all patients who have had at least two prescriptions for narcotics that have a single active ingredient, whose units are mg, and for whom the dosage of the ingredient in these prescriptions never decreased from one prescription to the next. Report their OHIP number.

10. Report the OCP number of the pharmacist who has had the greatest number of other pharmacists train under him or her.

11. For each pharmacist who has trained anyone, report their OCP number, the OCP number of the first person to complete training under them, and the OCP number of the last person to complete training under them. Your resulting relation should have three attributes: "OCP", "first" and "last".

12. Find all people who have, at least twice, had more than one prescription filled in a year, but haven't had one filled since 2014. Report the person's OHIP number and the last date on which they had a prescription filled.

Part 2: Additional Integrity Constraints

Express the following integrity constraints with the notation $R = \emptyset$, where R is an expression of relational algebra. You are welcome to define intermediate results with assignment and then use them in an integrity constraint.

1. A pharmacist can only train under someone who registered with the Ontario College of Physicians before they did.

2. A doctor can't prescribe a controlled substance (a product with schedule "narcotics") until after they have prescribed three different over-the-counter drug products (products with schedule "OTC").

When writing your queries for Part 1, don't assume that these additional integrity constraints hold, except for the first one — it was given in the schema as an integrity constraint.

Style and formatting requirements

In order to make your algebra more readable, and to minimize errors, we are including these style and formatting requirements:

- In your assignment statements, you must include names for all attributes in the intermediate relation you are defining. For example, write

$$HighestGrade(sID, oID, grade) := \quad \ldots$$

- Use meaningful names for intermediate relations and attributes, just as you would in a program.

- If you want to include comments, put them before the algebra that they pertain to, not after. Make them stand out from the algebra, for example by using a different font. For example, this looks reasonable:

  – Students who had very high grades in any offering of a csc course.
  $High(sID) := \Pi_{sID}\sigma_{dept='csc' \wedge grade>95}(Took \bowtie Offering)$

A modest portion of your mark will be for good style and formatting.

Submission instructions

Your assignment must be typed; handwritten assignments will not be marked. You may use any word-processing software you like. Many academics use LaTeX. It produces beautifully typeset text and handles mathematical notation well. If you would like to learn LaTeX, there are helpful resources online. Whatever you choose to use, you need to produce a final document in pdf format.

You must declare your team (whether it is a team of one or two students) and hand in your work electronically using the MarkUs online system. Instructions for doing so are posted on the Assignments page of the course website. Well before the due date, you should declare your team and try submitting with MarkUs. You can submit an empty file as a placeholder, and then submit a new version of the file later (before the deadline, of course); look in the "Replace" column.

For this assignment, hand in just one file: A1.pdf. If you are working in a pair, only one of you should hand it in.

Check that you have submitted the correct version of your file by downloading it from MarkUs; new files will not be accepted after the due date.