Exercise 1: Relational Databases

Introduction to Database Systems
Jorge-Arnulfo Quiané-Ruiz

Spring 2023

PostgreSQL installation

In order to run the exercise, as well many of the other exercises and homeworks of the course, you must install the PostgreSQL database system (postgresql.org) on your laptop. Installation guides that should work for 90+% of students have been posted on learnIT. It is best if you can do this *before* the exercise session, so that you can work on the exercise. If you have problems, you can bring them up to TAs during the exercise session.

1 Using PostgreSQL

Try downloading the database script from learnIT within Lecture 1. Create a database, then run the database install script, using psql, and then run the queries, both using pgAdmin and psql. Read through the queries and see whether you can make sense of them.

2 Your First Relational Database

Note: The code needed for this exercise is basically all in the slides, the emphasis here is on working with the database software. You should use this exercise to play with the SQL statements as much as you can.

Consider the sample Coffee database used in slides 60 onwards. Your task is to create part of this database. Write SQL commands to CREATE the following tables:

Coffees(<u>name</u>, manufacturer)

Coffeehouses(<u>name</u>, address, license)

Sells(<u>coffeehouse</u>, <u>coffee</u>, price)

The underlined columns should form the primary keys of the relations. Note that there should also be foreign key relationships in the definition of the last table to the corresponding columns in the first two tables. Note that slides 68-69 have commands for two of the three tables.

Maintain all the SQL commands in a script file, which you can run repeatedly using psql. Before creating a table, always drop the table if it exists (see slide 59). With foreign keys, you need to be careful about the order in which you drop tables (try dropping tables in the incorrect order!). Run this script from the command prompt—in the final exam you must be able to do this, so you might as well start early. Ask the TAs for help, if needed.

Once you have the basic database, try the following:

1. Write SQL commands to INSERT some data into the tables (see slide 75). Maintain these commands in a separate SQL file.

2. Try to create some simple SQL queries to SELECT data. (You can look to the queries developed in the demo on slide 21 for inspiration.) Maintain these commands in a separate SQL file.

3. Modify the database with some more advanced constraints (see slide 72). Then modify your INSERT and SELECT statements to test whether these constraints work.

Note: You should feel free to be creative with the last part, even to the point of creating some strange tables and test data. The more features you play with, the better.

3 A Simple Database—With and Without a DBMS

Note: If you do a lot of experiments with the coffee database above, you may not have much time for this last task. It is still a worthwhile thought experiment to do, when you have time. You should definitely do this in pairs or small groups.

Suppose that you must implement a system that contains information on the food in a canteen. The following information should be stored:

- Information on the dishes that can be bought. Each dish has a name and a price.
- Each day it is registered which dishes are for sale, and how many of each are sold.
- There is a list of ingredients, each having a name and a supplier. For each dish it is recorded what ingredients go in, and in what quantity.

a) How would you store this information *without using a DBMS*? Sketch an implementation in a language such as Java (or any other object-oriented, imperative language). Think about that the implementation should be flexible and e.g. allow that you change the supplier of milk. You are not expected to write any code, just give an overall design.

You can imagine many queries on such a database, e.g.:

- Find all dishes containing eggs.
- Find the total sales amount (in DKK) of today.
- Find the most sold dish of today.

b) How would you implement these queries? Sketch a solution. Especially for students who have taken the algorithms class: Think about how the solution scales to large data sets. Can you avoid repeated linear traversals of data?

c) Create a relational data model for the database. Write down the database schema.

Note: In the next two weeks we will see how the above queries can be done with little effort, and high performance, using a DBMS. You may re-visit this exercise towards the end of the class, to evaluate what you have learned.