Hand-in 3: Normal forms, JDBC, and indexing

The hand-in should be sent, by the same groups as in the first hand-ins, to Andrea Campagna, acam@itu.dk at the latest Monday October 20, 23.59.

Parts of the hand-in can be written by hand and scanned, but it has to be handed in electronically.

The hand-in will be corrected and you will get feedback from Andrea. The feedback is an important part of your learning process. It may contain comments on what to improve to get the hand-in approved. Hand-in 4 is due November 10.

The third part of the project builds on the previous two parts. In this part you will create the database (in MySQL unless we have agreed on something else), and use JDBC to populate it with information on movies and actors. You will implement a small program that uses queries from the second hand-in on the database.

Normal forms
Your database design should be in normal form (say, BCNF). If it is not, make changes to your E-R diagram and relation schemas. Part of the hand-in is to explicitly state all non-obvious assumptions needed to argue for normal form. For example, you may state assumptions such as "(movie,firstname,lastname) is a key for R". Your assumptions should be consistent with the data set provided.

JDBC
We provide you with a Java program and data files that can be used to load a small subset of IMDB into your database. You have the choice of two sizes, covering 3% and 10% of IMDB, respectively. Modify the program to load data in the right format, or load data in the default format, and use SQL to transform to data to your schema. For example, you may want to associate a unique identifier (integer) with each actor and each movie. (MySQL’s auto-increment feature comes in handy here.) To test your queries you will need to enter additional information (manually, or by making your own loader program).

Once data is loaded, you should make a small Java program (or a collection of programs) that acts as an interface to queries 1, 2, 4, 7, and 8 from hand-in 2. The program should prompt the user for the various constants in the queries (e.g. in query 1, you should be able to find the common movies of any two actors). The important part of the program is the use of JDBC, not the user interface – e.g. a text-based interface is fine.
Indexing
For each query above create one or more indexes that are used by the DBMS when executing the query. You should use “Explain query” in QueryBrowser to see that your indexes are used (or at least possibly used) for the queries. Discuss how the indexes can be used by the DBMS to evaluate the queries. Indicate for each query the observed speedup achieved by using the indexes compared to having no indexes.

To be handed in
- Name of all group members, including e-mail addresses
- An introduction describing the work
- A graphical presentation of the E-R model (possibly revised after the feedback meeting).
- A schema, written in SQL, corresponding to the E-R model.
- A list of assumptions about the data that implies that it is in normal form.
- The Java program implementing the interface.
- If you are not satisfied with some part of the hand-in, e.g. if you have discussed alternative solutions and don’t know which is the best choice, you should add some comments on it.

In addition, you should be prepared to demonstrate your program in connection with the exercises.

Course goals covered by this hand-in
After the course the students should be able to:
- Use SQL in applications (Java).
- Decide if a given index is likely to improve performance for a given query.