

# Introduction to Databases, ITU, Fall 2003

Rasmus Pagh

September 1, 2003

## Exercises on September 2

The goal of these exercises is to get crash started with (the web interface to) the MySQL database at ITU. By September 1st you should have received a MySQL user name and password by e-mail (otherwise contact sysadm). To start the system you go to the URL <http://mysql.itu.dk/myadmin>

1. Create a new relation (called a “table” in MySQL) called `Movie` with attributes as in G UW Figure 6.1 (use `CHAR(30)` as type for strings and `INT` as type for the other attributes), and insert the tuple (called a “row” or “record” in MySQL) on G UW page 241.
2. Run the SQL query: `SELECT * FROM Movie WHERE studioName = 'Disney' AND year=1990;` Explain the result, and add another Disney movie to the `Movie` relation which will **not** be returned by the SQL query.
3. Run the SQL query: `SELECT title, year%100 AS shortyear FROM Movie;` Explain what the `%` operator does.
4. Run the SQL query: `SELECT concat(substring(title,1,8),'...') FROM Movie;` Explain the result.
5. Direct your browser to <http://www.itu.dk/people/pagh/IDB03/data/> (and bookmark the URL). Run the SQL commands in the file `example.rel` to create the relation `example`.
6. Run the SQL query: `SELECT * FROM example WHERE t LIKE '%lo%';` Explain the result, and experiment with other patterns replacing `'%lo%'`. (Note that `%` has a different meaning in this context compared to above.)
7. In the following you are asked to write various expressions in SQL. You may write `SELECT *, X AS result FROM example;` to test the expression `X` on the sample data in the `example` relation. Replace `X` with each of the following:
  - (a) An expression that is true if the sum of `x` and `y` is more than 42.
  - (b) An expression that is true if exactly one of `a`, `b` and `c` is 1. (Hint: Use the boolean operators `AND` and `OR`).
  - (c) An expression that gives the first `z-y` characters of `t`.
  - (d) An expression that is true if `1900 + z` is a leap year.
8. Experiment with rewriting some of the SQL commands used above: First try to write variants of the queries. Are the results as you expect? Then play with the syntax: Introduce line breaks and spaces, remove parentheses, change from upper to lower case and vice versa. Whenever a change happens (relative to before), try to understand why. Change the commands such that MySQL does not accept them – read the error message.

**To be handed in no later than September 12, 11.59 AM:**

You may want to first read exercise 5.2.4 on page 210 for warship terminology. Hand in parts a, b, and d of exercise 6.2.3 on page 263 of G UW, plus the following addition:

- g) Find those ships whose class contains at least three ships. (Hint: Use three tuple variables.)  
Explain why each ship appears several times in the result.

Test your answers in MySQL by downloading the relations of Figure 5.12 from <http://www.itu.dk/people/pagh/IDB03/data/>