CSC 343H: Lecture Prep for Week 8

Task 1: JDBC Practice

This task will help ensure you are very ready to do the JDBC part of Assignment 2. If you are well into the JDBC part, you probably don't need to do this, so I am making Task 1 not for credit. (Task 2 alone will determine your prep8 grade.) If you are *not* well into the JDBC part of Assignment 2, I strongly recommend that you do Task 1 as a warm-up.

Before you begin: If you haven't already, go through the JDBC Exercise; this task is very similar.

For this exercise, we'll use the markus schema from Assignment 2 Login to CDF, and grab the starter code from the location `~csc343h/winter/public_html/prep/w8/Markus.java` (recall how to do this using `cp`, as in the JDBC exercise handout).

Implement the `main` function to do the following:

1. Prompt the user for the username of a student, with a string like "Enter the name of a student".
2. Read in from standard input the username of a student.
3. Query the database to find the username of everyone who has ever graded that student.
4. Print out the username of each of these graders.

In order for your program to report any graders back, you will need to put some data into your database. Create a file called `prep8-data.ddl` and put the necessary `INSERT INTO` statements to set up a scenario with

- 2 graders
- each has graded 2 groups (that makes 4 groups in total)
- there is one group of size 1, one of size 2, one of size 3 and one of size 4 You're welcome to use or adapt test data you've made for Assignment 2 (which you should have by now!).

Once you are convinced your Java program works properly, do the following to demonstrate it:

1. Start psql
2. Import the schema
3. Import your data in `prep8-data.ddl`
4. Exit psql
5. Compile and run your Java program

Capture all of this, including your commands and the output they generate, in a file called `prep8-demo.txt`. Don't hand any of this in. :-)

Task 2: Design a DTD

Design DTD rules for an XML file that will store a list of books. While it is up to you to decide what elements and attributes to define, here are the constraints you must enforce:

- A book has a title, one or more authors, and a year.
- A book can optionally have ratings, each of which is a number from 0 to 5.
- A book can optionally have reviews, each of which is text (perhaps very short, perhaps a few paragraphs). The username of the person who wrote the review can optionally be recorded along with the review.

Note: there are many perfectly acceptable ways to represent this data. Save your work in a file called `books.dtd`.

Create a file called `books.xml` that has your DTD rules embedded within it, is valid with respect to your DTD, and satisfies the following requirements:

- It must contain information on three books.
- Each book must have 2 ratings.
- One of the books must have a review.

These are not requirements to be enforced by your DTD. They are here just to make sure you create a dataset that is big enough to demonstrate your DTD design.

Run "xmllint --valid books.xml" to make sure that your XML file is valid, and save your interaction (including both the xmllint command and its output) in a text file called `prep8-demo.txt`. Then, submit ~~`books.dtd`~~, `books.xml` and `prep8-demo.txt` on MarkUs.