

## CS430/630 – Homework 4

Released Mar 29, Due Apr 13

**Instructions:** The homework is due on Fri, Apr 13, 23:59:59. Submission is done in electronic format, create a folder HW4 in your course directory and place source code and compilation scripts there. Recall to set the proper permissions for files!

### Question 1 (50 points) - JDBC

You must create a JDBC application for managing the course enrollment at a university. The schema is as follows:

```
Students(sid:integer, sname:string)
Courses(cid:integer, cname:string, credits:integer)
Enrolled(sid:integer, cid:integer)
```

The `Students` relation stores data about students: a unique student id and name. Each course has a course id, name and number of credits). The `Enrolled` relation stores what courses are taken by which students.

You must create the above schema definition in your submission in a file called `schema.sql`. You are allowed flexibility on the exact attribute types you use for your schema, as long as they reasonably match the specification above (e.g., in terms of number types, string types). Also, you have to create a Java JDBC-based application run by students, with name `Student.java`. The application must have a command-line interface menu that allows the user to select one option as below. Once that menu function is completed, the program must return to the main menu. For each menu option, you are allowed (and even recommended, if needed) to have multiple steps (or “screens”) to complete the tasks. You will use the DBS3 Oracle instance as DBMS.

#### Student Menu:

Application starts by requesting student’s ID; no authentication is necessary, and the remaining session assumes that student ID is active. If (-1) is introduced, a new student is created, and the user is prompted for all necessary information. The main menu is the following:

L – List: lists all records in the course table

E – Enroll: enrolls the active student in a course; user is prompted for course ID; check for conflicts, i.e., student cannot enroll twice in same course

W – Withdraw: deletes an entry in the Enrolled table corresponding to active student; student is prompted for course ID to be withdrawn from

S – Search: search course based on substring of course name which is given by user; list all matching courses

M – My Classes: lists all classes enrolled in by the active student.

X – Exit: exit application