CSI3317 / 3717— Database Management Systems, Fall 2005, University of Ottawa

Project: Database Application Development

Due date: Last day of classes (in the assignment box and electronically to a TA TBD)
Value: 15% of final marks
Note: to do in groups of 3 (or 4 in case of Management students)
Price: best project gets a price to be determined – probably a copy of a DB text book

---

In this assignment, you will apply much of the application-oriented material from Chapters 6 and 7 to create a small database-backed application. The focus of this project is to extend the conceptual and logical database design that you did in Assignment 1, to create a JDBC-based code that accesses a PostgreSQL database, and to create parts of the application that reside at the middle and presentation tiers of the three tier architecture seen in Chapter 7.

Description of the requirements — Recall the database design for a banking company called *Main Savings Inc.* that you gave for Assignment 1. Note that not every aspect of a realistic design was modelled. You rather considered a few aspects that illustrate some of the main ER design concepts seen in class. Below are the requirements we had for the banking database:

- Main Savings Inc. is organized into several branches. Each branch has a unique name and is located in a particular city. Moreover, each branch has an asset.

- Customers of Main Savings Inc. are known through a unique customer identifier, a first and last name, a street adress, and a city where they live. Customers must have a dedicated employee at the bank, who acts as that customer's banker. The latter supervises both the customer's borrowing of loans and the customer's possession of accounts. Such a supervisory role has a type (that may be a personal banker or a loan officer).

- Employees of Main Savings Inc. are identified by their employee ids. Moreover, the bank stores the first and last name, and the telephone number of each employee, and the employee's start date. Also, Main Savings Inc. keeps track of which employees works for which manager.

- Main Savings Inc. offers loans to its custommers. A loan originates at a particular branch and can be held by one or more customers. Each loan has a unique loan number, along with a loan amount. Main Savings Inc. keeps track of loan payments as follows: a loan payment number identify a particular payment for a specific loan; each payment has such a loan payment number, together with a payment date, and a payment amount.

- Main Savings Inc. offers both savings and checking accounts to its customers. Each customer can have more than one account and one account can be held by more than one customer. Each account is identified by an account number and has a balance, as well as the last date on which it was accessed by each owner. Finally, each savings account has an interest rate and each checking account has an indication of the allowed overdraft limit.

In addition to the requirements above, make the design more realistic by adding the following requirements:

- Main Savings Inc. maintains information about account activities. The latter are generated by customer accounts. each account activity indicates the account number, a date, a description (e.g. "payroll", "utility payment", etc.), an amount, and the remaining balance on the account.

- A customer of Main Savings Inc. has a phone number, an email, and the best time of the day when she is reachable.

- Main Savings Inc. offers its customers a facility for paying their utility bills online. A bill payment has an account from which the payment is made, a payee, an amount, and a payment date.

Based on the information given above, do the following.

1. **(10 points)** – Extend the ER diagram for the Main Savings Inc. database from Assignment 1 by taking the additional requirements into account.

2. **(10 points)** – Translate your ER diagram into a relational database schema by writing a PostgreSQL SQL script. Use appropriate constraints in your SQL script. Your script should run at SITE.

   (**Submit this script with your ER diagram by Oct 28**)

3. **(60 points)** – Write JDBC code that performs database access and manipulation. Assume that bank employees have a different interface than the one used by the customers. The interface to your system should offer a reasonable set of features including:

   (a) A user logs on in the system.

   (b) There is a menu with 4 items: Accounts, Payments, Tranfers, and Profile.

   (c) The Accounts menu offers 2 functionalities: Account Summary and Account Activity. Account Summary give a summary of (all) accounts held by a member, i.e. the account type and balance. Account Activity gives the account number, a date, a description (e.g. "payroll", "utility payment", etc.), an amount, and the remaining balance. The user should have the choice of showing the activities of either one of the accounts she owns.

   (d) The Payments menu offers an interface for paying bills. The user can specify the account to be used, the payee, the amount to pay, and the payment date.

   (e) The Transfers menu offers a simple interface for transfering money from a customer's account to another (possibly someone else's) account.

   (f) The Profile menu offers the possibility of changing a customer's email, telephone, and adress.

   (g) In addition to all the above features, bank employees have access to employees data. Find such a few appropriate queries for employee data and provide an interface for them. Such an interface will comprise meaningful search predicates.

   (h) The Java interface to your system should be simple and functional.

   (**Submit this by the last day of classes**)

4. **(20 points)** Furthermore, design a (simple) web-based presentation tier, and a middle tier using one of the technologies available: e.g. HTML pages, CGI scripts, Java servlets, JavaServer pages, or something else. Start from the code used for the Java-based part. Design your web pages by providing the features described above. Read Sections 6.6 and 7.8 to get an idea of how to design the architecture of your system.

   (**Submit this by the last day of classes**)

Some Resources: Links to material related to JSP, Servlets, XML, JDBC, HTML, etc:

http://www.cs.wisc.edu/~dbbook/openAccess/
http://www.java.sun.com/products/jdbc

http://www-db.stanford.edu/∼ullman/fcdb/oracle.html
http://www.utoronto.ca/ian/books/
http://www.postgresql.org/docs

To submit:

- Your ER diagram, SQL script, and Java code.

- Any further CGI code written or the like.

- A printout of the output of the system that clearly shows that what you did works.

You will give a short demo of your system to a TA.