**CS157A: Spring 2022**

**Project #1: Banking System (130 Points)**

**Due Date: Apr 22, 2022**

Description:

You are responsible to write a database application which acts as a simple banking system. This application must be able to do basic banking functions according to the specifications given below. This project must be implemented using DB2, Java, & JDBC.

Project Specification:

Section A: Schema Definition

P1.Customer (*ID*, Name, Gender, Age, Pin)

P1.Account (*Number*, ID, Balance, Type, Status)

-- All the attributes cannot be NULL.

-- Underlined attributes is denoted as the primary key of that relation.

-- Italicized attributes is system generated.

-- Attribute types and ranges:

- ID: integer (system generate starting from 100)

- Name: varchar(15)

- Gender: char (must be M or F only !!!)

- Age: integer (>= 0)

- Pin: integer (>= 0)

- Number: integer (system generate starting from 1000)

- Balance: integer (>= 0)

- Type: char (C for Checking, S for Saving)

- Status: char (A for Active, I for Inactive)

Section B: Data Administration & Manipulation

Screen # 1 (Title – Welcome to the Self Services Banking System! – Main Menu)

1. New Customer
2. Customer Login
3. Exit

For #1, prompt for Name, Gender, Age, and Pin. System will return a customer ID if successful.

For #2, prompt for customer ID and pin to authenticate the customer. If user enters 0 for both customer ID & pin, then you will go straight to Screen #4.

Screen # 3 (Title – Customer Main Menu)

1. Open Account
2. Close Account
3. Deposit
4. Withdraw
5. Transfer
6. Account Summary
7. Exit

For #1, prompt for customer ID, account type, and balance (Initial deposit). System will return an account number if successful.

For #2, prompt for account #, change the status attribute to ‘I’ and empty the balance for that account.

For #3, prompt for account # and deposit amount.

For #4, prompt for account # and withdraw amount.

For #5, prompt for the source and destination account #s and transfer amount.

For #6, display each account # and its balance for same customer and the total balance of all accounts.

For #7, go back to the previous menu.

Screen # 4 (Title – Administrator Main Menu)

1. Account Summary for a Customer
2. Report A :: Customer Information with Total Balance in Decreasing Order
3. Report B :: Find the Average Total Balance Between Age Groups
4. Exit

Note: The only way you can get to Screen #4 is by entering 0 as the ID and 0 as the pin in the customer login screen.

For #1, same function as #6 above except that you would need to input the customer ID explicitly.

For #2, you would display the customer ID, Name, Age, Gender, and total balance in decreasing order.

For #3, you prompt for a min & max age to compute and display Average Balance.

For #4, go back to the previous menu.

Section C: User Interfaces

1. Command line interface described in Section B.
2. If you have extra time, you can add GUI panels on top of the command line interfaces for possible extra credit.

Section D: Additional Notes:

1. The special administrator ID and Pin (0,0) can be hardcoded in your program as a special ID/PIN.
2. Customer IDs are system generated and initiated by the “New Customer“ operation.
3. The customer and administrator main menus are the default top level menu after an operation.
4. You can open an account for someone else but you cannot close someone else’s account.
5. You can deposit into other people’s accounts but you can’t withdraw from them.
6. You can transfer money from your account to someone else but not the reverse.
7. All the range checking need to be handled both in DDLs and your application.
8. Customer account summary should not include accounts in the “closed/inactive” state.
9. For all the administrator reports, closed/inactive accounts will not be part of the reports.
10. For database connections information, use a properties file.
11. Must handle error condition gracefully (e.g. should not crash and exit because of any exceptions).
12. You will be given Java code framework and you need to put your logic inside.