COMP 1451 Assignment 2

In this assignment we will focus on inheritance and testing.

Continuing from the ATM application in Assignment 1 we will build an inheritance structure for different kinds of accounts available at the bank. The class descriptions are as follows.

Account

The supertype. It will hold information common to all the subtypes. Fields include:

- double balance – the dollar amount in a subtype account
- String accountNumber – same as assignment 1 but no longer in the BankCustomer class.
- boolean active – true is the account is active, false when it is closed.
- ArrayList<String> for transaction information – every time there is a transaction a record of it will be stored here. This will apply to both deposits and withdrawals.
- void addTransactionInfo(String) is a method that will add a String that details the information when a deposit or withdrawal is mode in any of the subtype accounts.

SavingsAccount

A subtype of Account. It will hold information that is unique to a savings account. Fields include:

- final double MIN_AMOUNT – the minimum amount an account must have in order to remain active.

ChequingAccount

A subtype of Account. It will hold information for a chequing account. Fields include:

- final double FEE – a fee that will be charged for each cheque that is processed.
- Int numberOfCheques – the number of cheques processed during a given time period (one month for example).

GoldAccount

This is an account that is provided to senior citizens, people 65 years and older. Fields include:

- double interestRate – the monthly interest earned by the account
- boolean inOverdraft - used to determine if the account has a negative balance
- final double FEE – a monthly fee that is charge but only if the account is in overdraft
- final double OVERDRAFT_AMT - will be used to set the overdraft limit that account holders are allowed to withdraw.

All classes in the hierarchy will implement the default and overloaded constructors, getters, setters, toString methods. All validation will done in the setters which will be called from the overloaded constructor.

Changes to existing classes

BankCustomer

- Will no longer hold the accountNumber. That will be in the Account class now. Also to be removed will be the balance, which will now be held in the Account super class.
- Account myAccount - A customer will have a reference to an Account. This will be one of either the above mentioned subtypes. Be clear that the reference in this class is of the supertype.
- in age field and getter and setter will added to hold a customer's age. This will be used when setting up a GoldAccount
- All the rest remains unchanged, including the passcode.

Bank

The removeAccount method will be renamed “deactivate” it will now change the active setting on an account to false. It will not remove the account from the collection.

ATM

This class will have to be modified to incorporate all the new classes and functions added with Assignment 2.

- ATM will seed the Bank with different types of accounts.
- The subtype accounts will use a prefix code in the number. Examples are:
    - SA-123 for a SavingsAccount
    - CH-123 for a ChequingAccount
    - GL-123 for a GoldAccount
- prefix codes will identify the type of account when the account info is being displayed.
- As in Assignment 1 when the program is exited a summary of the current account will be displayed including all fees and interest amounts.
- The transaction record for the current account will also be displayed on exit.

Testing

A second part of Assignment 2 is to include JUnit tests.

- For this assignment only the Account hierarchy requires unit tests to be done.
- Testing will be thorough, using positive and negative tests.
- All the classes within the Account hierarchy need to be tested.
- Do not perform more than one test per unit.
- The test classes will be included in your submission.

Sample of summary after the user has exited the program.

```
Thank you for banking at Bullwinkle's Bank
ACCOUNT SUMMARY:
BankCustomer [firstName=Darby, lastName=Dog, passcode=123, account=SavingsAccount
[toString()=Account [balance=225.0, accountNumber=SA-123, active=true]], age=35]
Acount Activity:
Wed May 25 12:04:36 PDT 2016 - deposit: $200.00
Wed May 25 12:04:46 PDT 2016 - withdrawal: $75.00

DEBUG: Displaying all the accounts in the bank.
BankCustomer [firstName=Freckle, lastName=Cat, passcode=789, account=GoldAccount
[interestRate=0.025, overdraft=false, toString()=Account [balance=200.0,
accountNumber=GL-123, active=true]], age=65]
BankCustomer [firstName=Darby, lastName=Dog, passcode=123, account=SavingsAccount
[toString()=Account [balance=225.0, accountNumber=SA-123, active=true]], age=35]
BankCustomer [firstName=Myia, lastName=Dog, passcode=456, account=ChequingAccount
[numberOfCheques=0, toString()=Account [balance=50.0, accountNumber=CH-123,
active=true]], age=12]
```

Due the night before session 8 at 11:59pm. Submission format is the usual ZIP file that includes all the project files.