CS 2024 – ASSIGNMENT #5

Assigned: **9/26/2019**
Due: **10/2/2019**

PROBLEM:

This assignment is going to continue to "build" on our Menu assignment from A3 and A4. You will be able to re-use parts of your A3 Menu class and a little bit of your main.cpp. In short, you are going to create another Menu based system but this time your menu items will be commands to interact with a `Bank` class that you also create. Let's start again with an example run of the program:

I'll start by showing a sample run of what this should look like:

```
---- MENU ----
(1) New Account
(2) List Accounts
(3) Quit
Enter Choice> 1
NEW ACCOUNT:
ENTER ACCOUNT NUMBER:  100
ENTER ACCOUNT NAME:    Ron
ENTER OPENING BALANCE: 150
 ---- MENU ----
(1) New Account
(2) List Accounts
(3) Quit
Enter Choice> 1
NEW ACCOUNT:
ENTER ACCOUNT NUMBER:  101
ENTER ACCOUNT NAME:    Craig
ENTER OPENING BALANCE: 250
 ---- MENU ----
(1) New Account
(2) List Accounts
(3) Quit
Enter Choice> 1
NEW ACCOUNT:
ENTER ACCOUNT NUMBER:  102
ENTER ACCOUNT NAME:    John
ENTER OPENING BALANCE: 500
 ---- MENU ----
(1) New Account
(2) List Accounts
(3) Quit
Enter Choice> 2

ACCT# : NAME : BALANCE
100 : Ron : 150
101 : Craig : 250
```

```
102 : John : 500

 ---- MENU ----
(1) New Account
(2) List Accounts
(3) Quit
Enter Choice> 4
ERROR, unknown item selected
 ---- MENU ----
(1) New Account
(2) List Accounts
(3) Quit
Enter Choice> 3
Quitting!
```

How do we do this?

The Bank class has a very simple definition in its BankH.h header file:

```
class Bank
{
 public:
  void newAccount();
  void listAccounts();

 private:
  std::vector<BankAccount *> mAccounts;
};
```

It only does two things for this assignment. It stores a vector of dynamically allocated `BankAccount` instances (so you will re-use `BankAccount` from A4) and allows you to create a new account or list all the accounts currently stored in the vector. When creating a new account, you will prompt for an account number (`int`), name (`string`) and initial balance (`int`). Once obtained you will use those pieces of information to allocate (dynamically) a new instance of `BankAccount` and store it in the vector. When listing accounts, you will just iterate through the stored instance in `mAccounts` and print them out in a manner similar to what the sample run above does.

You may have already notice that you will have to make some modifications to `BankAccount` to store additional information (Bank Account Number, Name of Customer). You will need to provide *at least* getter functions to retrieve that information from the `BankAccount` class. Finally, you will need to modify and use the overloaded constructor that will now take three arguments (account number, customer name and initial balance). You will make use of this constructor when dynamically allocating an instance of `BankAccount`.

HINTS/DISCUSSION:

Pay attention to the sample printout above. It will help remind you what you should be doing!

You will need to (of course) modify MenuH.h and Menu.cpp again to account for the new commands. For this assignment, you can implement user interaction in member functions of the Bank class (that is, the newAccount and listAccounts methods can utilize cout/cin directly instead of leaving all of the “prompting” in global functions in main.cpp.

I’m purposely leaving out a number of details. Please feel free to use Piazza and/or office hours to ask questions!

FILES TO SUBMIT:

1. main.cpp
2. MenuH.h
3. Menu.cpp
4. BankAccountH.h
5. BankAccount.cpp
6. BankH.h
7. Bank.cpp
8. Writeup (.doc, .docx, .pdf, .txt accepted)