

The goal of this assignment is to get some practice writing classes with private fields, constructors, and methods to get and set fields as well as some other stuff. We also need to include a test driver to create objects and make sure they work correctly.

This assignment should be placed in package "hw01".

In this assignment, you will create a java class to keep track of a bank account, called the Account class. This class should have fields to keep track of the account number, and the current balance – the amount of money currently in the account. The account class should have a static "next account number" field, initialized to account number 1000. The creator method should take a single argument – a double precision floating point number which represents the initial balance in the account. (Use double precision just in case some really rich guy wants to open an account!) The creator method should assign the next account number to this account, and increment the next account number by 1.

The account class should also support the following methods:

`void deposit(double amt)` : A method that adds the amt value to the current balance. A deposit always works so no return value is required.

`boolean withdraw(double amt)` : A method that subtracts the amt value from the current balance. Note that account holders should not be able to withdraw more than the current balance. If the customer wants to withdraw more than the balance available, print out a message using the statement:

```
System.out.println("Account: " + <account number field> +  
    " requested a withdrawal of " + amt +  
    " but only " + <balance field> + " is available.");
```

In this case, withdraw the current balance in the account instead of the amount requested. This method should return "true" if the complete withdrawal was successful, and "false" if the withdrawal request could not be completely satisfied.

`void report()` : A method to report the account number of the bank account, and the current balance in the bank account using the statement:

```
System.out.println("Account: " + <account number field> + " Current balance: " +  
    String.format("$%.02f", <balance field>));
```

You should also create a class called Tester, which has a single main method. Your main method should create multiple bank accounts, make deposits and withdrawals from those bank accounts, and report on the status of those accounts. The main method should have enough invocations of account methods to make sure that your Account class works correctly.

Submission

Once all your Java code is written and tested, you will need to collect it all into a single file. Please create a zip file with the name “hw01_<userid>.zip”, where <userid> is your bmail userid. For instance, my bmail address is `tbartens@binghamton.edu`, I would create “hw01_tbartens.zip”.

Depending on what operating system you are on, there are different ways to do this:

- Unix or Mac OS: cd to the parent directory, then type:

```
zip -r hw01_<userid>.zip hw01
```
- Windows: Select the hw01 folder, then right mouse click, and select “sent to >”/“Compressed (zipped) folder”. You will then have to rename the resulting hw01.zip to add your userid.

When you have your Java code collected in a single file, submit that file on MyCourses under “Content/Homework Submissions/HW01”.

Grading

This assignment is worth 10 points.

- If there is a problem with your submission, such as the wrong file name or wrong archive format, but the code is in the submission, 2 points will be deducted.
- If your code does not compile, (i.e. if you get compiler errors) you will get eight points deducted.
- If your code produces any compiler warnings, 2 points will be deducted.
- If your code compiles, but produces incorrect results for *your* Tester program, 5 points will be deducted.
- The professor will write a Tester2 class with a main method that invokes your Account class methods. If you produce incorrect results using the professor’s Tester2 class, 3 points will be deducted.
- If you submit after September 8 at 11:59 PM, 2 points will be deducted. No credit will be given if you submit after September 11 at 11:59 PM.
- Your code will be compared against all other submissions. If your code is similar to another student’s (i.e. strong evidence that you both started from the same source), then both students will get a zero for this assignment.