Java Puzzle Ball MOOC
Lab 2: Static and Instance Variables

Overview

Lab 1 had a problem. All account owners had the same name. But good banking software should be able to accommodate the many different names and other properties of its customers when an account is created. Some properties differ for each individual instance, while other properties may apply to a class as a whole. Study the fields of the `SavingsAccount` and `CheckingAccount` classes. Carefully consider which fields which should use instance variables and which should use static variables. Modify the fields accordingly. Allow the initial values of instance variables to be set through the constructor. Set the initial values of static variables when its field is declared. Make any other necessary changes and test your work from the `main` method.

Tasks

Download the MOOC's Lesson 2 lab (JPB_MOOC_Lab2.zip). This zip contains a NetBeans project with the start state for your program. This project is slightly different from the Lab 1 solution. Unzip the file and open the project in NetBeans. The program contains `SavingsAccount`, `CheckingAccount`, `TestClass` classes, all of which are half-complete. Your goal is to complete all three files.

Savings Account

This bank has one type of `SavingsAccount` which earns 2% interest. The account should also keep track of a customer's name, balance, account number, and their account type.

Checking Account

A `CheckingAccount` is similar to a `SavingsAccount`. However, a `CheckingAccount` has no interest rate and is therefore incapable of earning interest.

Test Class and Main Method

Play with the `main` method to test the changes you make to the `SavingsAccount` and `CheckingAccount` classes. You may need to remove some of the values previously passed to the constructors.

- Create several instances of each account type.
- Observe the values these instances.
- If two people coincidentally have the same name, will their accounts at least have different balances?