



Core

Deadline: Sunday of Week 2

Difficulty Level: Intermediate

Est. Time: 2 – 4 hrs

BankAccount Assignment



Learning Objectives:

- Practice member variables
- Practice instance methods and getters
- Implement static variables and methods

We are going to create a BankAccount class. This class will recreate some of the common account transactions that normally occur for a bank account, such as displaying your account number, checking and savings amount, total amount. Of course, there are also methods to invoke, such as depositing a check, checking your balance, and withdrawing money.



To get started, we will create the following two files:

BankAccount.java

```
public class BankAccount {
    // MEMBER VARIABLES
    private double checkingBalance;
    private double savingsBalance;
    private static int accounts;
    private static double totalMoney; // refers to the sum of all bank account checking
and savings balances
    // CONSTRUCTOR
    // Be sure to increment the number of accounts
    // GETTERS
    // for checking, savings, accounts, and totalMoney
    // METHODS
    // deposit
    // - users should be able to deposit money into their checking or savings account
    // withdraw
    // - users should be able to withdraw money from their checking or savings account
    // - do not allow them to withdraw money if there are insufficient funds
    // - all deposits and withdrawals should affect totalMoney
    // getBalance
    // - display total balance for checking and savings of a particular bank account
}
```

Java

<

Java OOP

>

Java OOP

▲

- [Overview](#)
- [Objects and Classes](#)
- [Member Variables](#)
- [OOP Quiz 1](#)
- [Orders & Items](#)
- [Getters, Setters & Access Modifiers](#)
- [Constructors and Overloading](#)
- [this](#)
- [More on Methods](#)
- [OOP Quiz 2](#)
- [Barista's Challenge](#)
- [Static](#)
- [BankAccount](#)

BankTest.java

```
public class BankTest {
    public static void main(String[] args){
        // Create 3 bank accounts
        // Deposit Test
        // - deposit some money into each bank account's checking or savings account and
        display the balance each time
        // - each deposit should increase the amount of totalMoney
        // Withdrawal Test
        // - withdraw some money from each bank account's checking or savings account and
        display the remaining balance
        // - each withdrawal should decrease the amount of totalMoney
        // Static Test (print the number of bank accounts and the totalMoney)

    }
}
```

Review the starting code and be sure to test your classes thoroughly.

- ☐ Create a BankAccount class with the correct member variables and methods.
- ☐ Users should not be able to set any of the attributes from the class.
- ☐ Create a BankTest class to test each BankAccount method you have created.
- ☐ Ninja Bonus 1: Add the following class attribute: account number. The type is up to you, can be a String or a Long or another type.

Submit



1705600698245-bankaccount.zip

Successfully submitted on January 18, 2024



[Previous](#)

[Next](#)

[Privacy Policy](#)