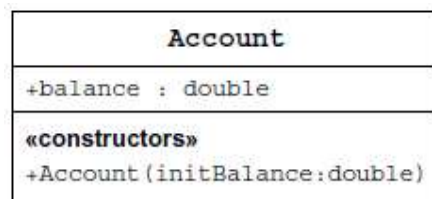


Object-Oriented Programming

Lab 2: Exploring Encapsulation V1

In this exercise, you explore the purpose of proper object encapsulation. You create a class in two steps to demonstrate the use of information hiding. In this version, you create an Account class with public data members. You will then create a test program that demonstrates the danger of using the public data directly.

Figure below shows the UML class diagram of the Account class that you will create in this exercise. This class will have one public data member (or instance variable), called balance, that maintains the monetary value of the customer's bank account.



There is only one business rule that applies to the Account class; the balance of the bank account must never go below zero. In this exercise, you will discover that the Account class cannot ensure this business rule (but you are going to fix this issue in the next exercise).

Note: For this exercise, please comment all methods (getBalance, deposit, and withdraw) to hide it.

1. Load your BankProject if it is not already opened.
2. Add an instance variable balance in the Account class. Initialize the balance instance variable with the parameter of the constructor.
3. Add the test class called TestAccount2 with a main method. This class acts as a program to create an Account object with an initial balance of 100. The test program will then add 47 and then subtract 150. Finally, the test program must print out the balance of the object to the standard output stream.

The output of the TestAccount2 program should be similar to the following.

```
Final account balance is: -3.0
```