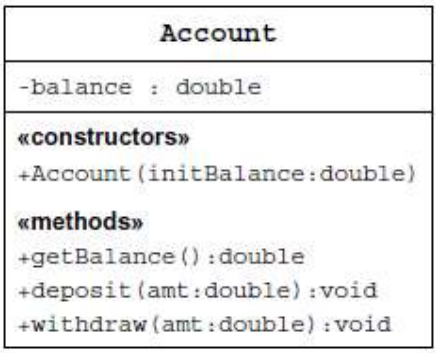
Lab 3: Exploring Encapsulation V2

In this exercise, you explore the purpose of proper encapsulation. You modify the Account class to hide its data member and provide public methods to manipulate the balance. You then use the test program that you have created in Lab 2 to test that business rule (balance must not go below zero) is satisfied.

Figure below shows the updated UML class diagram of the Account class that you have created. This design for the Account class hides the instance variable, balance.



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

1. Open the Account class and modify the source file. This class must satisfy the UML diagram above.
2. In this task, modify the TestAccount2 class to call methods on Account to deposit 47 to and withdraw 150 from the Account object.
3. Run the test class.

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

```
Final account balance is: 147.0
```

The 150 withdraw command did not take effect, because it would have made the balance drop below zero. However, the Account object did not tell program that the withdraw command failed, it ignored the command. You will fix this problem in future exercises.