You have been given a template for the Currency and Money classes, with JavaDoc comments explaining what each method should do. There is also a Bank and Account class, but we will come back to that in the following section. All the methods of Cur- rency.java and Money.java are empty.

First, write test cases for the methods of each class, and then fill in the methods with code that will make your test cases pass. The template test methods are a guideline you can use for constructing your tests. Unless you have a good idea for how to otherwise structure the tests, you should simply fill in those template test methods.

Important note: The decimal numbers representing money are implemented as integers. The last two digits denote two decimals. So Money(200050, SEK) will mean 2000.50 SEK.

**Assignment Testing Methodology**

Bank

The Bank and Account classes were written by a bad programmer. When you are confident your Money and Currency classes work as intended, write test cases for the Bank and Account classes and find the bugs. Again, the specification is provided in the JavaDoc comments.

ToDo Checklist

1.Write the code for the body of the test case methods of MoneyTest.java and CurrencyTest.java. Remember all these tests will be now in failed state since the actual methods are not yet implemented.

2.Motivate your test cases by commenting in the test methods.

3. Now complete the methods body in Money.java and Currency.java classes, and comment where necessary. Again make sure you follow the specification for each method and implement the required functionality.

4. If you run your test cases for MoneyTest.java and CurrencyTest.java at this point, they should be all pass.

5.By following the Bank.java and Account.java specification in the JavaDoc comments, write your test cases in the BankTest.java and AccountTest.java files. Note which of your test cases fail, by commenting in the corresponding test meth- ods.

6.Find the bugs in Bank.java and Account.java based on the JUnit failures from the previous step, and note it by placing a comment in the code where the bug was spotted. Explain how you found the bug with your unit tests.

7.Fix the bugs and verify that BankTest.java and AccountTest.java passes suc- cessfully.

8. Above unit tests were not strictly unit tests per se. They have been working like integration tests. So use Mockito to mock the dependencies and do strictly UNIT Testing.