

# About Testers

1. The idea of a tester is that it thoroughly tests your code without having manually create objects, call methods on them, etc.:
2. The tester should have a method that runs all tests (it may call other methods to run individual tests)
  - a. If a test fails, it should print an error message identifying the expected vs actual result and return immediately
  - b. If the run-all-tests methods completes without error, it should print a message indicating success
3. A test object (an object of the class being tested) should be constructed, using either a local variable in the run-all-tests method or a field in the tester class. The latter will keep the object being tested around for later examination if needed.
4. The tester should initialize the test object with data needed for the tests. For example, if removing zero-balance accounts, there should be at least one such account present.
5. The class being tested should not be modified to make private structures available- you may create other methods to get data about them, such as returning the size of a private ArrayList<>
6. Tests of methods returning an ArrayList<> should capture the returned value in a local variable and interrogate it.
7. If statements should be used to check against expected values
8. get() should be used to check ArrayList results by element, you should at least check against an ArrayList's expected size
9. You cannot simply print Object types like ArrayLists: unless a toString() method is defined for it, attempting to do so will print out the class name and an 8-digit hexadecimal number. Instead, use the methods defined for that Object type to retrieve useful information for printing.
10. To test printing methods that do not return a value, print a string explaining what to expect the method being tested to print, then call the method
11. You do not need to call a method being tested, then call it again to get a return value
12. It should be immediately clear to an observer whether the tests completed successfully or not without having to walk through your code.