CS583: Object-Oriented Testing
Assignment 5
StockWatcher
Points: 10
Due: 10/31/2017

Objective: The student will gain experience in planning and writing tests using a testing framework.

Assumptions:

1. This is an team assignment. Each team member is to submit the solution.
2. Use Eclipse.

Background:

The project manager asks you to test code (StockWatcher). This application is not a critical application. This code was written by someone on the development team. You will test the code using JUnit and include your test results in a document to be emailed to the QA manager for his report. **(NOTE: Describe the test you are conducting and if it is supposed to pass/fail and why.)**

Procedure:

1. Download StockWatcher.zip and unzip on your PC. Import the project into Eclipse.

2. Before you start writing your own unit tests for StockWatcher, make sure the components of the test environment are in place. You can do that by running StockWatcherTest.java which will execute the starter test, testSimple.

3. Notes:

    a. Like all GWT JUnit test cases, the StockWatcherTest class extends the GWTTestCase class in the com.google.gwt.junit.client package. You can create additional test cases by extending this class.

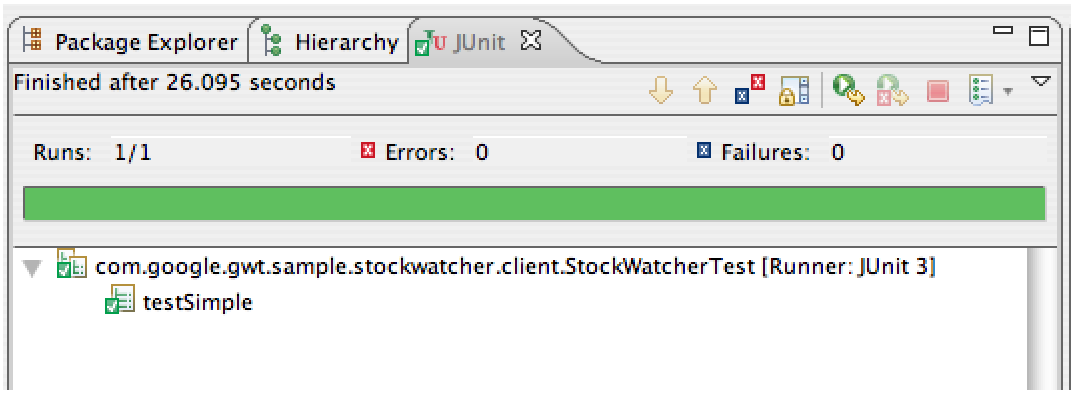
    b. The StockWatcherTest class has an abstract method (getModuleName) that must return the name of the GWT module. For StockWatcher, that is com.google.gwt.sample.stockwatcher.StockWatcher.

    c. The StockWatcherTest class is generated with one sample test case, testSimple. This testSimple method uses one of the many assert* functions that it inherits from the JUnit Assert class, which is an ancestor of GWTTestCase. The assertTrue(boolean) function asserts that the boolean argument passed and evaluates to true. If not, the testSimple test will fail when run in JUnit.

4. Run unit tests **in Eclipse (using the Google Plugin for Eclipse)** The Google Plugin for Eclipse makes it easy to run tests in Eclipse.

5. Run the JUnit test in development mode.

a. From Package Explorer, right click on the test case you want to run, select Run As > GWT Junit Test
b. The simpleTest executes without error.



```
Finished after 26.095 seconds
Runs: 1/1    Errors: 0    Failures: 0
com.google.gwt.sample.stockwatcher.client.StockWatcherTest [Runner: JUnit 3]
    testSimple
```

6. Write a JUnit test to verify that the constructor of the StockPrice class is correctly setting the new object's instance fields.
   a. To the StockWatcherTest class, add the testStockPriceCtor method as shown below.

```
/**  * Verify that the instance fields in the StockPrice class are set correctly.  */
public void testStockPriceCtor() {  String symbol = "XYZ";  double price = 70.0;
double change = 2.0;  double changePercent = 100.0 * change / price;

StockPrice sp = new StockPrice(symbol, price, change);   assertNotNull(sp);
assertEquals(symbol, sp.getSymbol());   assertEquals(price, sp.getPrice(), 0.001);
assertEquals(change, sp.getChange(), 0.001);   assertEquals(changePercent,
sp.getChangePercent(), 0.001); }
```

   b. Rerun StockWatcherTest in development mode.
   c. Both tests should pass.

```
[junit] Running com.google.gwt.sample.stockwatcher.client.StockWatcherTest
[junit] Tests run: 2, Failures: 0, Errors: 0, Time elapsed: 16.601 sec
```

Optional:
Write positive and negative *tests of value*.

Deliverables:
1. Everyone: submit to Assignments link for Assignment **5: StockWatcher**
   - list the bugs found
   - showing screenshots and describing what the test is intended to do and why each test passes or fails
2. Indicate your responses to the following objectives:
   1) what is your test plan? write a bulleted list that consists of some ideas you have on testing this code
   2) how many positive tests did ***you*** write?
   3) how many negative tests did **you** write?

3. Read http://java.dzone.com/articles/unit-testing-private-methods. Also, read http://www.artima.com/suiterunner/privateP.html.

Optional:

1. Which method would you use in order to test private methods for this assignment?
2. Output test results in a file containing descriptive test case name with its associated test result: PASS, FAIL.

Contact Info:

Larry McCartney
Click the Email function.