

Unit Testing with JUnit

The goal of this lab will be to perform unit testing on your BuddyInfo and AddressBook classes using the JUnit framework. As mentioned in class, JUnit happens to be included in Eclipse, so you won't need to download it and use it outside your development environment.

Part 1- Some more reading!

There is a very good article available on cuLearn:

test-driven-development-paper.pdf

It explains, in detail, how to use Test-Driven Development (TDD) to write code. Read the paper carefully and implement their example as you go along. You will be required to submit the source code for this when you are finished. Note: The article was written assuming a previous version of Java and JUnit. Keep the following issues in mind while working through the article:

- The article uses JUnit 3. Either import the JUnit 3 library or, if you prefer JUnit 4, the only change you will need to make is in the AllTests main method. Change the code in that method to "junit.textui.TestRunner.run(AllTests.suite());"
- The article was written before Java supported generics so you will get warnings related to your ArrayList. Modify the code slightly to get rid of those warnings.
- The article, when showing the code for the test classes, calls one of the classes TestEmptyMovieList when it should be TestMovieListWithEmptyList.

Part 1b - Converting JUnit 3 to JUnit 4

To convert `TestMovieListWithEmptyList`, `TestMovieListWithOneMovie`, `TestMovieListWithTwoMovies`:

1. Change scope of setUp() method to public from protected
2. Add @Before before the setUp() method.
3. Remove extends TestCase on all test classes
4. Add @Test before each test method
5. Since the test classes do not inherit from TestCase anymore you have to use the static assert methods provided from the Assert Class

To Convert AllTest:

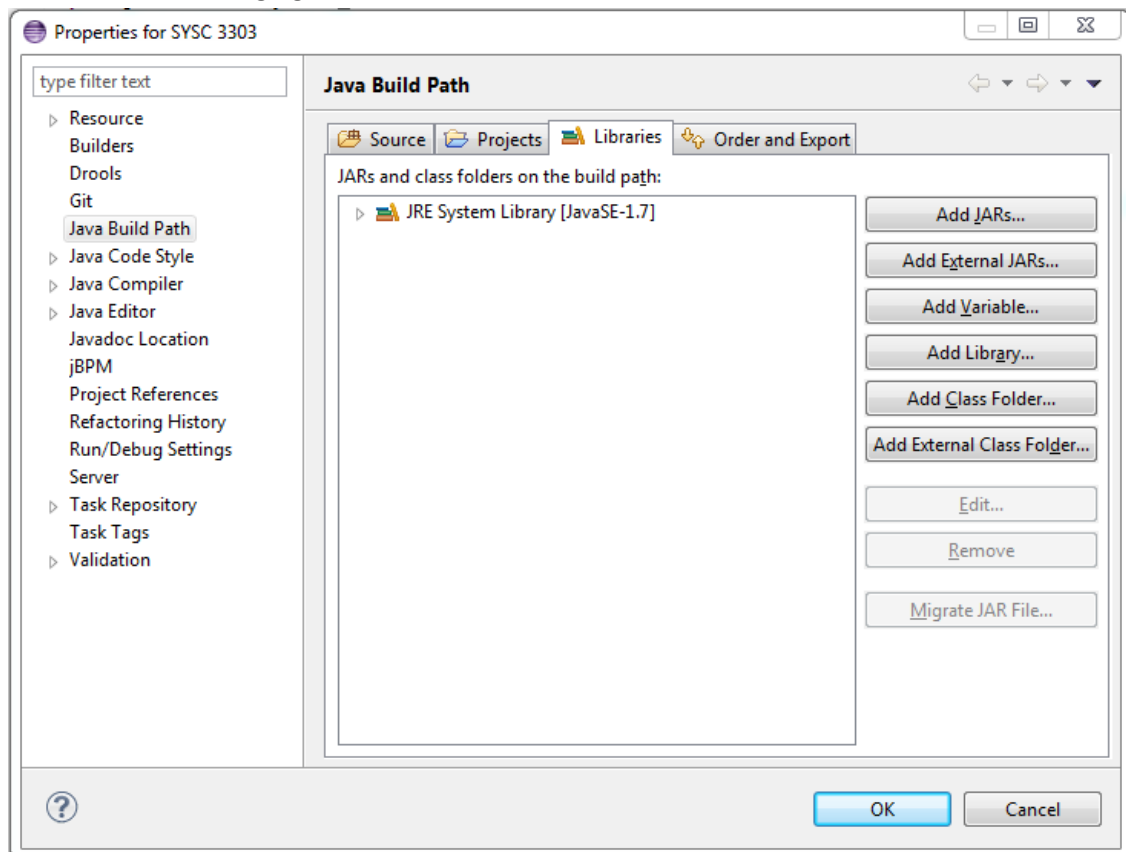
1. Remove extends TestCase from AllTest class
2. Remove all methods from AllTest class
3. Add @RunWith(Suite.class)
4. Add @SuiteClasses({ /* Add list of classes of test here */ })

More information can be found : <http://www.vogella.com/tutorials/JUnit/article.html>

Part 2- Setting-up JUnit for your project

To be able to use JUnit in Eclipse, you need to add it to your “build path”, i.e. tell the Java compiler where to find the code for that framework.

1. Open the project that contains your BuddyInfo application
2. Right-click on the project name and select “Properties”
3. Go to “Java Build Path” and then select the “Libraries” tab. You should have something that looks like the following figure:



4. Click on “Add Library”.
5. Select “JUnit” and click the “Next” button.
6. On the next screen select JUnit 4 and click “Finish”.
7. You should now be back to the Project Properties panel, and the JUnit JAR file is added to your build path. Click OK.

Now you’re ready to create your first unit tests! The easiest way to create test case classes is to use the JUnit wizard:

1. Right-click on the file for which you want to create test cases, say BuddyInfo, and select New->JUnit Test Case. If JUnit Test Case is not listed, select Other and then locate JUnit Test Case from the list.
2. In the next panel, accept all the defaults (but note that for your group project you should consider putting your unit tests in their own package). In addition, click the options to create

setup() and tearDown() methods if you think you will need them for the given class under test. Click Next.

3. In the next panel, you will be offered the possibility to create empty methods to test each of the methods in the given class under test. Select the ones that correspond to the test cases you had in mind (and of course you will have many more anyway!) and click Finish.

Part 3- Test away!

The test class is now generated; it is now up to you to populate it! My advice is that you should first write down on paper in plain English the test cases that you would like to implement: what behavior should be tested, what should be the outcome?

A few ideas:

- Test every public method!
- What happens when your methods are given bad parameters?
- Have you tested your methods with a range of values?

Basically, you want to think of any possible way to break your code! Running the JUnit tests in Eclipse is similar to running a Java application. First, make sure that the test case class you want to run is selected, then from the menu, select: Run ->Run As->JUnit Test. That should open the JUnit frame and will display test failures as well as the infamous red bar which you want to turn green.

Also, if you select your project and click Run->Run As->JUnit Test from the menu it will run all of the tests in your project.

You are required to add the following methods (and test them!):

- In BuddyInfo: A constructor that takes a BuddyInfo object as a parameter and creates the BuddyInfo with the same values as the parameter (a copy constructor).
- In BuddyInfo: A method that returns a String that contains a greeting for your buddy.
- In BuddyInfo: Methods that set and get your buddy's age. You should also have a method isOver18() that returns a Boolean value.
- In AddressBook: A size() method that returns the number of Buddies in your AddressBook.
- In AddressBook: A clear() method that empties your AddressBook.

Deliverables:

- The source code from the article you read.
- The source code of the classes BuddyInfo, AddressBook, AddressBookTest and BuddyInfoTest.