TDD with JUnit Lab 1

Here are the Using TDD, develop a simple system to Each quote is for a single requirements: calculate course costs student for students of a college Base cost is given Other factors by this table affecting cost: Length cost Residency: Out of 500 3 weeks state students pay 4 weeks 700 10% additional 6 weeks 1100 8 weeks 1500 School loyalty: 12 weeks 2000 \$50 off per year of enrolment

TDD with JUnit Lab 1

Use the following conventions for your tests – examples:

- TargetClass is tested by TargetClassTest
- TargetClassTest is in same package as TargetClass
 - But use separate source folders for target and test code src/com/company/project/TargetClass.java test/com/company/project/TargetClassTest.java
 - Allows you to test package private methods, yet still keep target and test code separate

Testing may require exposing some things not required by the client (but try to keep this to a minimum)

- You may need to think beyond what the user client needs, to incorporate what the test needs
 - Different method signature, package private methods, etc.

TDD with JUnit Lab 1 (end of lab)

Write the tests first (and make sure you see them fail first)

Think about how you will test cost calculation and the desired API

Follow the TDD cycle: red-green-refactor

Write the test class first, with all the tests

Write the actual class, focus on making the bar green

Fill in the methods with their actual functionality

Take appropriate step sizes

Err on the side of smaller steps

You can always increase step sizes as you gain confidence and learn about the code you are writing