TCSS 305 Programming Practicum
Assignment 2: Junit Testing for Rentz Login/Registration
Value: 5% of the course grade
Due: <mark>Monday, 3 Feb 2019, 23:59:00</mark>

Program Description:

Assignment 0, Assignment 1, 2, 3, 4 aim to develop Rentz, a vehicle rental system. You will incrementally build your code for each assignment, i.e., your Assignment 2 will be built using the code that you submitted for Assignment 1. Make sure you finish every assignment on time, so that your future submissions remain unaffected. Assignment 2 focuses on writing Junit Test Cases for the User and Registration classes developed in Assignment 1.

Using Previous Code: Rename you project to "username-rentz-2" and share it to the repository following steps 1-3 under "Using Previous Code" given in "hw1.pdf".

Implementation Guidelines: You need to create Junit 4 test cases for the methods in User and Registration classes using the guidelines given below. NOTE: You need to add a method and make changes to an existing method in Registration class as mentioned below.

`User`:
- Write test cases for all the methods of the class.
- The test class should be under a new package called `tests`
- All scenarios including exceptions thrown should be tested in detail
- You can use multiple JUnit test methods to cover all testing scenarios for a single method in `User` class
- Test classes should have proper java docs
- Assert statements should have proper Assertion error messages
- All test cases should pass
- The coverage of test cases for `User` class should be 100%
- If you have extra utility methods (they should be private), they need NOT be tested

`Registration`
- Write test cases for all methods in the registration class except `printSignin`
- The test class should be under the package called `tests`
- All scenarios including exceptions thrown should be tested in detail
- You can use multiple JUnit test methods to cover all testing scenarios for a single method in the `Registration` class
- Test classes should have proper java docs
- Assert statements should have proper Assertion error messages
- All test cases should pass
- The coverage of test cases for `Registration` class should be >40% (100% not necessary) since we are not testing `printSignin`
- The extra utility methods, such as extra credit in Assignment 1, should be private methods and you do NOT need to write testcases for these methods

Submitting your executive summary:

The executive summary template will be provided to you in canvas. Download the file. Rename the executive summary "`executivesummary-username-rentz-2`", where `username` is your UWNetID. Fill and upload the *executive summary* to canvas as the same WORD format. The grader will grade your submission based on the executive summary as well as the project you submit to the SVN repository. If you do not submit either of them, you will receive no grades.

Grading Rubric (Total 100 points):

Task	Max Score Possible
Executive Summary -Submission on canvas with correct version number	2
Source Code – Submission to SVN	5
User test class – correct private fields and setup method	2
User test class – parameterized constructor1 functionality tested	5
User test class – parameterized constructor1 exceptions tested	2
User test class – parameterized constructor2 functionality tested	5
User test class – parameterized constructor2 exceptions tested	2
User test class – GetMyName functionality tested	2
User test class – GetMyPassword functionality tested	2
User test class – GetMyVIPStatus functionality tested	2
User test class – toString functionality tested	5
User test class – equals functionality tested	5
User test class – hashcode functionality tested	2
User test class – all test cases pass	5
User test class – 100% code coverage	5
Registration test class - correct private field and setup method	2
Registration test class – parameterless constructor functionality tested	5
Registration test class – login functionality tested	5
Registration test class – login exceptions tested	2
Registration test class – register functionality tested	5
Registration test class – register exceptions tested	2
Registration test class – toString functionality tested	5
Registration test class – clear functionality tested	1
Registration test class – getMyUserList functionality tested	1
Registration test class – all test cases pass	5
Registration test class ~ >40% code coverage [all functions except printSignin should be tested	5
Correct Project Name and Executive summary name	2
No Errors and Warnings (except reasonable ones which should be clearly explained in executive summary)	5
Proper Java docs and Header comments	4
Total	100