Lab 2

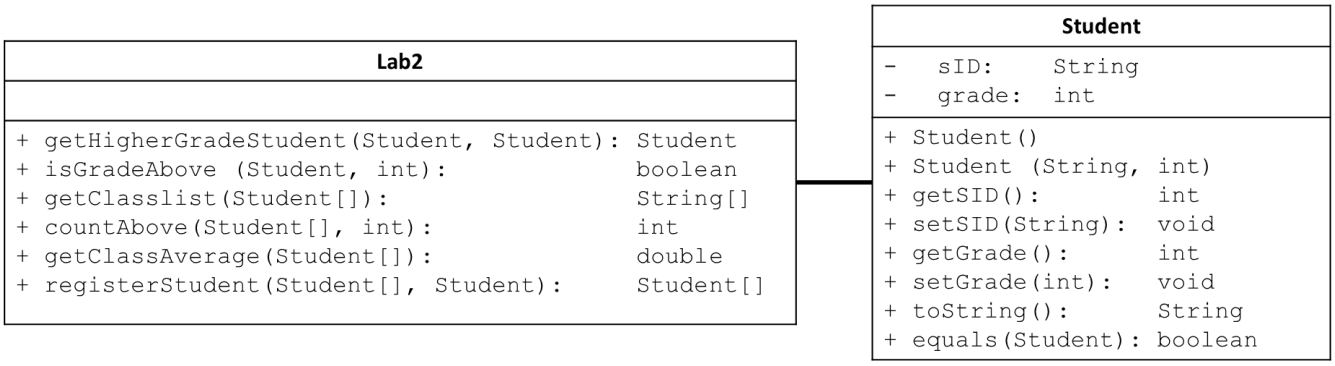
Reminder: Your code is to be designed and written by only you and not to be shared with anyone else. See the Course Outline for details explaining the policies on Academic Integrity. Submissions that violate the Academic Integrity policy will be forwarded directly to the Computer Science Academic Integrity Committee. All materials provided to you for this work are copyrighted, these and all solutions you create for this work cannot be shared in any form (digital, printed or otherwise). Any violations of this will be investigated and reported to Academic Integrity.

Objectives

- Exposure to testing
- Practice writing static methods that take Objects and arrays of Objects as parameters

Part I

1. Download Lab2Tester.java, Lab2.java and Student.java to your Lab2 working directory.
2. Compile and run Lab2Tester.java. There should be no errors.
3. Complete the methods in Lab2.java and overviewed in the following UML diagram.
    a. The documentation for each method is in Lab2.java. Read the documentation provided before each method to understand what we are asking you to implement.
    b. There are some initial tests written in Lab2Tester.java for some of the more difficult methods found in Lab2.java. The tests provided provide additional insight on how we expect each method to be implemented
    c. Write and test each method **one at a time**.



CHECKPOINTs (Ungraded) – if after reading through the documentation and tests for any of the methods you are required to implement you are still unsure what the method is supposed to do, don't hesitate to ask a TA for an explanation of what is expected.

CHECK POINT 1: getHigherGradeStudent and isGradeAbove
CHECK POINT 2: getClasslist and countAbove
CHECK POINT 3: getClassAverage and registerStudent

Finished early in the lab? Start your Assignment!

SUBMISSION (Graded) – Submit the **Lab2.java** and **Lab2Tester.java** files into the Lab2 submission page on BrightSpace.