## Lab 2

### Preliminaries

The name of your Eclipse project should be abc123-lab2, where you replace abc123 with your abc123 id. The name of the file that contains your main method should be Lab2.java. You will also need a Grades.java file for constructing and processing Grades objects. To submit the project, export the project and upload the zip file to Blackboard.

### Task

The task is to read a file, store the data into objects, and process the objects. The file is formatted as pairs of lines: the first line of each pair is the name of a student, and the second line is a list of grades. This data will be stored in a Grades object. The Grades class will have several methods for processing the data.

For example, if this is the contents of the file being processed:

Alice

87 99 96 99 86 96 77 95 70 88

Bob

73 78 76 80 99 96 73 96 76 78 78 92 93 75 93

Camila

99 94 85 99 99 93 81 95 76 80 77 79 98 72 98 97 92

Diego

76 97 72 92 86 86 89 85 81 87 76 80 89

then the following should be printed:

Alice [87, 99, 96, 99, 86, 96, 77, 95, 70, 88]

Name: Alice

Length: 10

Average: 89.30

Median: 91.5

Maximum: 99

Mininum: 70

Bob [73, 78, 76, 80, 99, 96, 73, 96, 76, 78, 78, 92, 93, 75, 93]

Name: Bob

Length: 15

Average: 83.73

Median: 78.0

Maximum: 99

Mininum: 73

Camila [99, 94, 85, 99, 99, 93, 81, 95, 76, 80, 77, 79, 98, 72, 98, 97, 92]

Name: Camila

Length: 17

Average: 89.06

Median: 93.0

Maximum: 99

Mininum: 72

Diego [76, 97, 72, 92, 86, 86, 89, 85, 81, 87, 76, 80, 89]

Name: Diego

Length: 13

Average: 84.31

Median: 86.0

Maximum: 97

Mininum: 72

### Reading in the Data

Assume that the name of the file is data.txt. In Eclipse, the file needs to be in the top directory of the project. Reading all the data in the file can be accomplished by a sequence of code like:

Scanner in = null;

try {

in = new Scanner(new File("data.txt"));

} catch (FileNotFoundException exception) {

System.err.println("failed to open data.txt");

System.exit(1);

}

while (in.hasNext()) {

String name = in.next();

while (in.hasNextInt()) {

int grade = in.nextInt();

}

}

Of course, you need to add code so that Grades objects are created with the data that is read.

### Testing

Each Grades object you create should be tested with the following static method.

public static void testGrades(Grades grades) {

System.out.println(grades.toString());

System.out.printf("\tName: %s\n", grades.getName());

System.out.printf("\tLength: %d\n", grades.length());

System.out.printf("\tAverage: %.2f\n", grades.average());

System.out.printf("\tMedian: %.1f\n", grades.median());

System.out.printf("\tMaximum: %d\n", grades.maximum());

System.out.printf("\tMininum: %d\n", grades.minimum());

}

This method should be included in your Lab2.java file exactly as it appears. testGrades should be called on each Grades object after all of its data has been stored in it. Note that this code requires that you implement several methods within the Grades class. Note: the toString method should return a String and should not contain any print statements within it. See the example output above.

### Comments

Create javadoc comments for your Java classes and your methods.

### Rubric

* An incorrect submission will possibly get zero points.
* (90 pts.) The output is correct (no extraneous output).
* (10 pts.) There are javadoc comments: at least one for the class and another for the main method.

Revision Date Tue Aug 19 2014 11:21:58 GMT-0500 (Central Daylight Time)