

| |
|--------------------------------|
| Program 5 - Due Nov. 25 |
|--------------------------------|

(Total: 100 pts)

Write a noninteractive Java program (program should not expect any information from the user) titled "YourLastName_Grading" that reads lines of input from a file with information about grades from a class. The input file consists of eleven lines with the following information:

- Student ID Number, the student's identification number
- Extra Credit, an integer representing the number of extra credit assignments turned in
- Grade, the percentage grade earned by the student

Valid student ID's are integers in the range from 00000-99999 and must be 5 characters long, extra credit is an integer in the range from 0-5 and represents a 1% increase of the student's earned grade to compute the enhanced grade, and the grade is a real positive number in the range 00.00 to 100.00.

The program will read in lines as input from the text file, accumulate summary data about the students while ignoring any line with any invalid data. The summary data will include information on the mean, mode, minimum, and maximum of the grades (both the earned grades and the enhanced grades), including the student ID of the students with the highest and lowest grades (both earned and enhanced).

*NOTE: Do **NOT** use arrays, or other topics we have not yet taught. Use enough variables to save all the information for each student.

For example:

The following shows an example input file:

```
51673 0 98.85
19438 5 95.00
00483 3 73.16
P1905 1 85.61
80463 2 73.16
76049 4 63.48
34086 7 90.23
13157 0 54.34
24937 2 81.03
26511 1 74.16
20034 4 103.90
```

The following shows an example interaction captured in a file by the command "% script Grading.out"

```
Script started on Tue Sep 24 10:23:58 2013
% java Diaz_Grading
```

```
Ignoring student with invalid ID number P1905
Ignoring student 34086 with invalid extra credit 7
Ignoring student 20034 with invalid grade 103.90
```

```
Mean Earned Grade: 77.15
Mean Enhanced Grade: 79.15

Mode Earned Grade: 98.85
Mode Enhanced Grade: 75.16

Minimum Earned Grade: 54.34 (student 13157)
Minimum Enhanced Grade: 54.34 (student 13157)

Maximum Earned Grade: 98.85 (student 51673)
Maximum Enhanced Grade: 100.00 (student 19438)
```

% exit

script done on Tue Sep 24 10:24:17 2013

Implement the following methods in the program:

- public static void validateFile(File inputFile) – this method will read in the file location, if the input file does not exist it will give an appropriate error message and terminate the program.
- public static void validateData(File inputFile) – this method will validate the data from the input file, if the data is valid it will save it appropriately and have the program ignore any lines of data which are invalid.
- public static double mean(Boolean gradeType) – this method computes and returns the mean of all grades. If the passed boolean is false, it will compute for earned grades only, otherwise it will compute for enhanced grades.
- public static double mode(Boolean gradeType) – this method computes and returns the mode of all grades. If the passed boolean is false, it will compute for earned grades only, otherwise it will compute for enhanced grades.
- public static double min(Boolean gradeType) – this method computes and returns the minimum of all grades. If the passed boolean is false, it will compute for earned grades only, otherwise it will compute for enhanced grades.
- public static double max(Boolean gradeType) – this method computes and returns the maximum of all grades. If the passed boolean is false, it will compute for earned grades only, otherwise it will compute for enhanced grades.
- public static void printData() – this method will display the results to the screen, and uses the printf method to align the data as shown above.

Run the program with the provided input file and capture all interaction in a file using the script command.

What to turn in:

- Soft copy of the results using the script command
- Soft copy of the program (the .java file submitted to Blackboard)