COP-2250 Java Programming

Programming Assignment 4

FIU Knight Foundation School of Computing & Information Sciences

For this programming assignment, you will implement of a java program for storing grades of students in a class, finding the average of grades, the maximum and minimum grade, and finally, printing the name and grade of a student given their ID number.

Program Input Handling

The program must first ask the user to enter the information of the first student via keyboard by printing the following message: After user enters the requested information (e.g. “Ann Smith 1234567 93”), the program must repetitively ask for information of the next student by printing the following message in a while loop:

```
Welcome to my grade book!
Please enter the information of the first student using the following format:
“firstName lastName PID grade”.
Press Enter when you are done.
```

```
Please enter the information of the next student using the same format.
If there is no more students, please enter the keyword “DONE”.
Press Enter when you are done.
```

We assume that the user input meets the following criteria (no need to do error handling):

- firstName is a single word and does not include any white space character.
- lastName is a single word and does not include any white space character.
- PID is a seven digit integer with no leading zeros.
- grade is a non-negative integer that doesn’t exceed 100.

Data Structure of the Program

For keeping students' information, you need to have four arrays defined in the main method of your program in the following way:

```
String[] firstNames = new String[STUDENTS_LIMIT];
String[] lastNames = new String[STUDENTS_LIMIT];
int[] pIDs = new int[STUDENTS_LIMIT];
int[] grades = new int[STUDENTS_LIMIT];
```

where STUDENTS_LIMIT is a final static variable defined in the only class of your program containing the main method and is initialized to 100 as we assume that the number of students for this program does not exceed 100:

```
public static final int STUDENTS_LIMIT = 100;
```

Static Methods of the Program

Other than the main method, your program needs to have the following methods:

```
public static double findAverage(int[] grades, int numberOfStudents){
    //Your code comes here!
}
public static int findMinimum(int[] grades, int numberOfStudents){
    //Your code comes here!
}
public static int findMaximum(int[] grades, int numberOfStudents){
    //Your code comes here!
}
public static int findGrade(int pID){
    //Your code comes here!
}
public static String findFullName(int pID){
    //Your code comes here!
    //This method returns the full name (first + “ ” + last)
}
```

Program Command Handling

After the user enters the information of all students one-by-one and enters the keyword “DONE”, your program must keep asking for a new command (e.g. print the message “Please enter a new command”) and respond to each command properly. Here are the list of all commands that your program must support:

- *average*: Your program must calculate and print the average grade of all students as the response to this command.
- *min*: Your program must calculate and print the minimum grade of all students as the response to this command.
- *max*: Your program must calculate and print the maximum grade of all students as the response to this command.
- *grade XXXXXXX*: Your program must find and print the grade of the student whose PID is given by the command (XXXXXXX is the PID where each X represents a digit).
- *name XXXXXXX*: Your program must find and print the full name of the student whose PID is given by the command (XXXXXXX is the PID where each X represents a digit). Full name is made of first name, followed by a single white space and the last name.
- *quit*: Your program must stop asking for more commands and quit.

Submissions and Grading Criteria

You need to submit a single file named "GradeBook.java". Here are the grading criteria:

- Code readability: 5%
- Using comments to explain every line of the program: 5%
- Correctness of Java syntax (no compilation error): 5%
- Correct implementation of each of the five methods (excl. the main method): 10% each, 50% total.
- Proper usage and naming of variables: 5%
- Proper implementation of the main method including the input-handling phase (when the students' info are stored in the arrays) and the command-handling phase (when the program parses each command, calls appropriate methods, and prints the results): 30% (15% for input-handling and 15% for command-handling)