**CPSC 24500: Student Database**

**Summary:** Using our previous student database assignments and the examples provided in class and on blackboard, you’ll add a Student class. Rather than work with just the variables, you’ll create a student object. You’ll assign the data to that student object and write the data out to a database file (a CSV file that will act as our database).

When our user selects to Write Data To File, it will append the students information like so:

Name,year,gpa

John, Senior, 3.8

Joe, Sophomore, 4.0

Phil, Freshman 4.0

Sarah, Junior 3.5

You will also add the ability to search for a student by name (this isn’t provided, you’ll add it). When/if that student is found store all the data in a Student object and print the data from that. Something like: *System.out.println(Student.name);*.

**\*IMPORTANT\* data duplication is a real world problem. What do we do if there are 3 John Smiths? At this time we won’t need to solve this problem. You can take 1 of 2 approaches here to get full credit. Assume there are 2 John Smith’s in the studentdatabase.csv file. 1) Print out the data for both. 2) Print out the data for the first one found.**

**Requirements**

*Interface:*

Same as week 1 with the following changes.

Menu:

1. Enter The Students Name:
2. Enter the Students Academic Year:
3. Enter the Students GPA:
4. Print Current Working Student
5. Write Data to File
6. Read Data From File
7. **Search by Student Name see extra credit**
8. **Delete Student see extra credit**
9. Exit

As with week 2 user cannot write to database until they’ve answered all the questions.

**Grading Criteria:**

12 pts total.

Menu prints as displayed below **1pt** **note if you do extra credit option 8 will be delete**

Menu:

1. Enter The Students Name:
2. Enter the Students Academic Year:
3. Enter the Students GPA:
4. Print Current Working Student
5. Write Data to File
6. Read Data From File
7. Search by Student Name
8. Exit

All questions can be answered, answers are stored and logically displayed as necessary **4pts**

Print Current Working Student will print the student you’re currently updating **see search** **2pts**

Search By Student Name will find a student by name and store it as your current working student **2pts**

Read data from File will print all the students in the database (just displays what is in the database) 1**pt.**

Write Data To File writes the current working student to a file. Note, you must use the student object you create to write to the file. You must also ensure you append to the file. The database should grow as you add students. The file format must be CSV and must match the format listed in the summary. **2pts**

**Extra Credit**:

2 pts – Find a way to handle duplicates. If I search John and 10 John’s exist how do you know which one I want? Note, just presenting 1, or a random John isn’t enough. This is a design/creativity challenge. How can you handle duplicate data?

2 pts – Delete student by name. I’ll enter a name and you’ll delete that student. NOTE, this is related to duplicate data. If there are 10 Johns, and I enter John to delete, I only want to delete one.

Solve both of these and I’ll give you an additional 2 pts. There are a few ways, and I’ve seen some interesting choices. Both must handle duplicates correctly, only allowing me to delete 1 based on input I provide. How you define that behavior is up to you.