

CS256 Lab 2 - Have some class...

Value: 25 points

Due: 6 September, 2019 @ 23:50

Submission Format: Two files, the python source code and the output file.

In this lab you will be practicing with Python's class definitions and related bits. As before you will be editing files remotely, connecting to tools.cs.earlham.edu, running your Python script from the command line on tools, and processing command line arguments.

Tasks

- 1) Develop a class named **Student** with following arguments and attributes:
 - a) **ID**: stores the unique ID number of a student.
 - b) **firstName**: stores the first name of a student.
 - c) **lastName**: stores the last name of a student.
 - d) **email**: stores the email address of a student.
 - e) **year**: stores which year (FR, SO, JR, SR) the student is in.
 - f) **gpa**: stores student's GPA in float.
 - g) **Getters and setters**: getters and setters for each of the class attributes. (get_email, etc.)
- 2) Write a function **loadStudents(inputFileName)** that takes the input file name as an argument and returns an array of populated Student objects. This can be done by iterating over the input and for each line instantiating & populating a Student object and then storing that object in an array which is ultimately returned to main().
- 3) Write a function **displayStudents(studentArray)** that takes an array of Student objects and uses the getters to print their contents using the getters to stdout in CSV format, like so:
ID, firstName, lastName, email, year, gpa
...
- 4) Write a function **averageGPA(studentArray)** that takes an array of Student objects and uses the getters to calculate and print their average GPA rounded to 2 decimal places to stdout.
- 5) Write a function **improveGPA(studentArray, percentImprovement = 1.0)** that traverses an array of Student objects and uses the getter & setter functions to increase every student's GPA by percentImprovement amount.
- 6) In your main() function, process the command line arguments and then invoke your functions in this order: **loadStudents()**, **displayStudents()**, **averageGPA()**, **improveGPA()**, and **averageGPA()** (with the default argument for percentImprovement). From the command line redirect the output into a file named **studentsWGPA.csv**. Your output file should contain the list of students with the two average GPAs at the end.

Implementation Notes

- The input file name should be the one required command line argument
- Check to make sure the file exists before calling **loadStudents(inputFileName)**
- We will provide a CSV formatted data file of students, etc. for you to use in testing