Assignment #4: Classes & Fundamental Data Types

*Due: Friday, October 26th @ 11:59PM*

*Total Possible Points: 20*

How to Submit

* Moodle assignment (no emails or hardcopies accepted)
* Submit IDE project in ZIP or RAR format as *Assignment4.zip* or *Assignment4.rar*

Goals

* To write secure, user-centric programs that validate input based on requirements.
* To use decision structures and Boolean operators to write more complex programs.
* To write a predicate method within a class.
* To document your code effectively and generate Javadocs for your classes.

|  |  |
| --- | --- |
| Forbidden | If you have trouble getting your Javadoc tool to work, look at this help page:  *Generating Javadocs:*  <https://jpgrady28.azurewebsites.net/Home/Docs/235> |
|  |  |

## Your Task

The majority of you are at or near the point in your academic careers where you need to think about doing an internship. Are you ready to do an internship? Do you even know if you are *eligible* for an internship? Knowing is half the battle, kids…[[1]](#footnote-1)

**Your task is to create a** Student **class that determines a student's eligibility for an internship based upon their GPA and number of credits earned. You will then write an** InternshipApp **class that tests the** Student **class's methods**. Your test program asks the user to enter his/her name, number of credits earned, and GPA. Your output clearly tells them his/her class standing (first-year, sophomore, junior, or senior), as well as if he/she is eligible.

**Additionally, your program must validate SEPARATELY what the user entered for his/her number of credits and GPA**. **This includes**:

* Testing for values that are out of range (e.g., entering -4.00 or 8.25 for GPA)
* Testing for data type mismatches (e.g., the user enters non-numeric characters for GPA or number of credits.)
* Alerting the user when the input was bad, then setting the GPA or number of credits to 0

Your classes must meet the requirements laid out in the grading rubric.

## Program Output

Here are a few examples of what the program's output should look like. Your instructor will test your output with several input scenarios to ensure all logic is correct:

*Test 1: All input valid*

Please enter your name:

Savannah

How many credits have you earned? 78

What is your GPA? 3.63

Savannah, you are a Junior who is eligible for an internship.

*Test 2: Good number of credits, bad GPA (value out of range)*

Please enter your name:

Jarrod

How many credits have you earned? 90

What is your GPA? 7.50

Bad GPA. Setting GPA to 0.00

Geoff, you are a Junior who is not eligible for an internship.

*Test 3: Bad number of credits (value out of range), bad GPA (type mismatch)*

Please enter your name:

MAKE ME

How many credits have you earned? -2000000000

Bad number of credits. Setting credits to 0

What is your GPA? the stuff of legends

Bad GPA. Setting GPA to 0.00

MAKE ME, you are a First-year student who is not eligible for an internship.

## Student Class Design

These are the methods your Student class must implement. All methods are public. *See the Grading Rubric for the class's instance variables*.

|  |  |  |  |
| --- | --- | --- | --- |
| Method name | What it does: | What info it needs: | What info it returns: |
| Student | Constructs a Student object | * Student's name * Number of credits * GPA | N/A |
| getClassStatus | Determines and returns the student's class status (first-year, sophomore, junior, or senior) depending on the student's number of credits | Nothing (*information available internally*) | Class status |
| isEligibleForInternship | Determines and returns the student's eligibility for an internship | Nothing (*information available internally*) | True/False: is the student eligible? |

|  |  |
| --- | --- |
| Lightbulb | **HINTS on the** Student **class's methods**:  getClassStatus : do you know when you're considered a Sophomore, Junior, or Senior? If not, here you go:   * First-year student: 0 to 29 credits * Sophomore: 30 to 59 credits * Junior: 60 to 89 credits * Senior: 90 or more credits   isEligibleForInternship : see the footnote on page 1. Read and apply the knowledge you gain to coding this method. |

## Requirements/Grading Rubric

* YES = Full credit
* Partial = Half credit
* NO = No credit

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Implemented**  **Successfully?** | | |  |  |
| **Requirement** | **YES** | **Partial** | **NO** | **Max Points** | **Comments** |
| ***Program Setup (3 points)*** | | | | | |
| Has a package named school |  |  |  | **1** |  |
| Has a class named Student |  |  |  | **1** |  |
| Has a class named InternshipApp with a main() method |  |  |  | **1** |  |
| ***Student class (6 points)*** | | | | | |
| Has three (3) private instance variables with appropriate data types:   * name: stores the student's name * credits: stores the number of credits earned * gpa: stores the student's GPA |  |  |  | **1** |  |
| Has one (1) constructor that initializes the student's name, credits, and GPA with the given parameter values |  |  |  | **1** |  |
| Implements the getClassStatus method with correct logic, parameters and return types; uses a multiple alternative decision block in its logic. |  |  |  | **2** |  |
| Implements the isEligibleForInternship as a predicate method with correct logic, parameters and return types. |  |  |  | **2** |  |
| ***InternshipApp test program class (9 points)*** | | | | | |
| Declares variables for the student's info with appropriate data types and relevant, descriptive variable names |  |  |  | **0.5** |  |
| Constructs a Scanner object to read console input |  |  |  | **0.5** |  |
| Prompts for and stores a name, number of credits, and GPA |  |  |  | **1.5** |  |
| Checks that the value entered for the number of credits is the right data type |  |  |  | **0.5** |  |
| Checks that the entered number of credits is within an acceptable range (0 to 150); alerts the user if it does not and sets credits to 0. |  |  |  | **1** |  |
| Checks that the value entered for the GPA is the right data type |  |  |  | **0.5** |  |
| Checks that the entered number of credits is within an acceptable range (0.00 to 4.00); alerts the user if it does not and sets GPA to 0. |  |  |  | **1** |  |
| Constructs a Student object using the entered name, credits, and GPA |  |  |  | **1** |  |
| Calls a method of the Student class to determine internship eligibility |  |  |  | **1** |  |
| Prints correctly the student's name, class standing (using a Student class method), and internship eligibility |  |  |  | **1.5** |  |
| ***Documentation (2 points)*** | | | | | |
| Javadoc-style code comments for the Student class and all public methods, including constructors |  |  |  | **1.5** |  |
| Generates full Javadocs for the project |  |  |  | **0.5** |  |

1. <https://www.thomas.edu/career-alumni/professional-and-career-development/learn-about-internships/> Hit the "Student Internship Guidelines and Requirements". [↑](#footnote-ref-1)