**IC10**

**Classes**

You have been chosen to develop a new Learning Management System for GGC. We need to track several types of data. First, we need to track the sections of the courses that are being taught. Second, we need for each section to track the students that are assigned to each section. Your job is to create a Java program that will consist of three classes.

# Section class

The first class should model a section. Information that must be tracked is:

* Section Number an int – Must be between 1 and 15 inclusive
* Course title a String – For this program, we are only going to allow Programming Fundamentals or Intermediate Programming.
* Instructor a String – Cannot be blank
* Class room a String – Cannot be blank
* Students an ArrayList of Student

The Section class should provide appropriate constructors with getters and setters for each attribute. You should create an addStudent method rather than the getter and setter for Students. These cannot be static and must be instance variables. All instance variables should be private. You should create a toString() method which will return a String that displays the information in the Section including the Students in the class. Your setters and the four-argument constructor should display appropriate error messages if the Section doesn’t pass validations. Since we have not done exceptions yet, you can display messages in your 4-argument constructor in the Section class. You should handle these errors in your driver when calling a setter.

# Student

Each student will track the following information:

* Name a String – Cannot be blank
* Major a String – Cannot be blank
* GPA a double – Must be greater than 0 and less than or equal to 4.00
* Hours an int – Must be greater than 0 and less than or equal to 17

The Student class should provide appropriate constructors and getters and setters for each attribute. All variables should be private instance variables. In addition, you should provide a toString() method which prints the student name, major GPA and number of hours. This class should validate each entry. Again, since we do not have exceptions, your constructor in Student can display the message but the test class should handle this error and enforce proper entry.

# SectionDriver

Finally, you should create a driver class that allows the user to enter the Course information and once the course information is entered, allow the user to enter students. When the user enters an empty String for the student name, your program should not prompt for more information for the student and then display the information entered by the user.

You must submit three files for this assignment. One containing the Section logic., one containing the Student logic and the final being your test driver.

Please enter the section number (1 to 15)

0

Invalid section. Please enter the section number (1 to 15)

16

Invalid section. Please enter the section number (1 to 15)

15

Please enter the course name

Course must be either Programming Fundamentals or Intermediate Programming

Please enter the course name

A prog

Course must be either Programming Fundamentals or Intermediate Programming

Please enter the course name

programming fundamentals

Please enter the instructor’s name

Invalid instructor name. This cannot be blank

Please enter the instructor’s name

Dr. Fred Smith

Please enter the room for programming fundamentals

Invalid room. This cannot be blank

Please enter the room for programming fundamentals

W2131

Please enter the student's name

Sally Jones

Please enter the major for Sally Jones

IT

Please enter the GPA for Sally Jones

5

GPA must be between 0 and 4, inclusive

Please enter the GPA for Sally Jones

-1

GPA must be between 0 and 4, inclusive

Please enter the GPA for Sally Jones

4

Please enter the hours for Sally Jones

0

Hours must be between 1 and 17, inclusive

Please enter the hours for Sally Jones

18

Hours must be between 1 and 17, inclusive

Please enter the hours for Sally Jones

15

Please enter the student's name

Tom Johnson

Please enter the major for Tom Johnson

Major cannot be blank

Please enter the major for Tom Johnson

Accounting

Please enter the GPA for Tom Johnson

3.5

Please enter the hours for Tom Johnson

12

Please enter the student's name

Section Course Inst Room

15 programming fundamentals Dr. Fred Smith W2131

**Name Major GPA Hours**

**Sally Jones IT 4.0 15**

**Tom Johnson Accounting 3.5 12**

Rubric:

Code standards – 10

Section

Instance variables – 5

Validation – 10

Student ArrayList – 5

Constructor – 10

addStudent – 10

toString – 10

Student

Instance variables – 5

Validations – 5

toString – 10

Tester

Create course – 5

Create students – 5

Output – 10