

Program #3
COP 2551_12004
Spring 2011

Due: Wednesday, March 9th, start of class.

Drop dead date: March 14th (we are on Spring Break then)

Objectives:

- Gain experience using multiple classes
- Gaining experience with Java programs of multiple classes / objects
- To gain experience in the detailed design of Java classes as well as Constructors, ToString, and other functionally-required methods.
- To gain additional experience with loops and conditional expressions.
- To gain experience with method calls and parameter passing
- To gain experience with instance variables and class variables.
- To gain experience in seeing association relationships and specifically aggregation relationships and how to implement these. UML diagrams must reflect the associations / aggregation relationships.
- To gain experience using DecimalFormat class and charAt method of String.

Requirements:

This program has much to do about objects and their dependencies and relationships. You will be creating a number of objects, performing some calculations, sending messages, use ToString to print out values of objects, and formatting output.

First of all, to accomplish much of this program, you will need to study/review class variables (static variables), DecimalFormat class objects, charAt method of String objects, UML examples, and more.

This program deals with students and their classifications (freshman, sophomore, etc.), the class that they are enrolled in, their gpa, and other summary statistics.

You are to develop a class called StudentBody. Your 'Main' class, and in particular, your main() method will be responsible for maintaining a number of Courses and Students in these courses.

You will instantiate six Student objects. For each Student object, you are to have instance variables: name, classification, number of hours taken, and grade points. Name and classification are strings, with the classification range "freshman" ... "senior" and "graduate". Hours taken and grade points are integers.

You will also create a `Course` class and build three course objects. For each `Course` object, you are to include instance variables course number, time of offering and two students enrolled in that course. Specifically, each `Course` object will be created with two student objects in it. Course number and time of offering are strings, while students are objects of type `Student`. You will create three courses, each of which as two students. Input data is presented ahead.

Please note that this program is VERY similar to one in your book and in my slides. Please use that as a guide to understand this specification and to help develop your design and implementation.

Inputs

So that we are all dealing with the same data, create your objects using this data ahead (you may 'hard code' these entries into your program at this time. Things will significantly change once we have had arrays. (This code may appear in `main()`)

```
// create a number of Student objects
Student stu1 = new Student ("George_Washington", "senior", 120, 375);
Student stu2 = new Student ("John_Adams", "senior", 110, 320);
Student stu3 = new Student ("Thomas_Jefferson", "grad", 80, 240);
Student stu4 = new Student ("James_Madison", "junior", 82, 230);
Student stu5 = new Student ("James_Monroe", "junior", 64, 148);
Student stu6 = new Student ("John_Q._Adams", "grad", 70, 275);

// creates two Course objects via declaration.
Course course1 = new Course ("cop2551", "MW 6pm", stu4, stu5);
Course course2 = new Course ("cis4328", "MW 3pm", stu1, stu2 );
Course course3 = new Course ("cap6100", "MW 7:45", stu3, stu6);
```

You will need to design the `Student` class and the `Course` class. You will need Constructors and a `ToString()` in each and maybe more.

Outputs

Your output should appear as follows: Use the formatting indicated below.

```
=====
Course1:  cop2551  MW 6pm

Student Name: James_Madison
Classification: junior
Hours Taken: 82
Grade Points: 230
Grade Point Average is: 2.8
```

Student Name: James_Monroe
Classification: junior
Hours Taken: 64
Grade Points: 148
Grade Point Average is: 2.31

=====
Course2: cis4328 MW 3pm

Student Name: George_Washington
Classification: senior
Hours Taken: 120
Grade Points: 375
Grade Point Average is: 3.12

Student Name: John_Adams
Classification: senior
Hours Taken: 110
Grade Points: 320
Grade Point Average is: 2.91

=====
Course3: cap6100 MW 7:45

Student Name: Thomas_Jefferson
Classification: grad
Hours Taken: 80
Grade Points: 240
Grade Point Average is: 3

Student Name: John_Q._Adams
Classification: grad
Hours Taken: 70
Grade Points: 275
Grade Point Average is: 3.93

S t a t i s t i c s

Total Number of courses offered is: 3

Number of graduate courses is: 1

Number of undergraduate courses is: 2

Total number of students is: 6