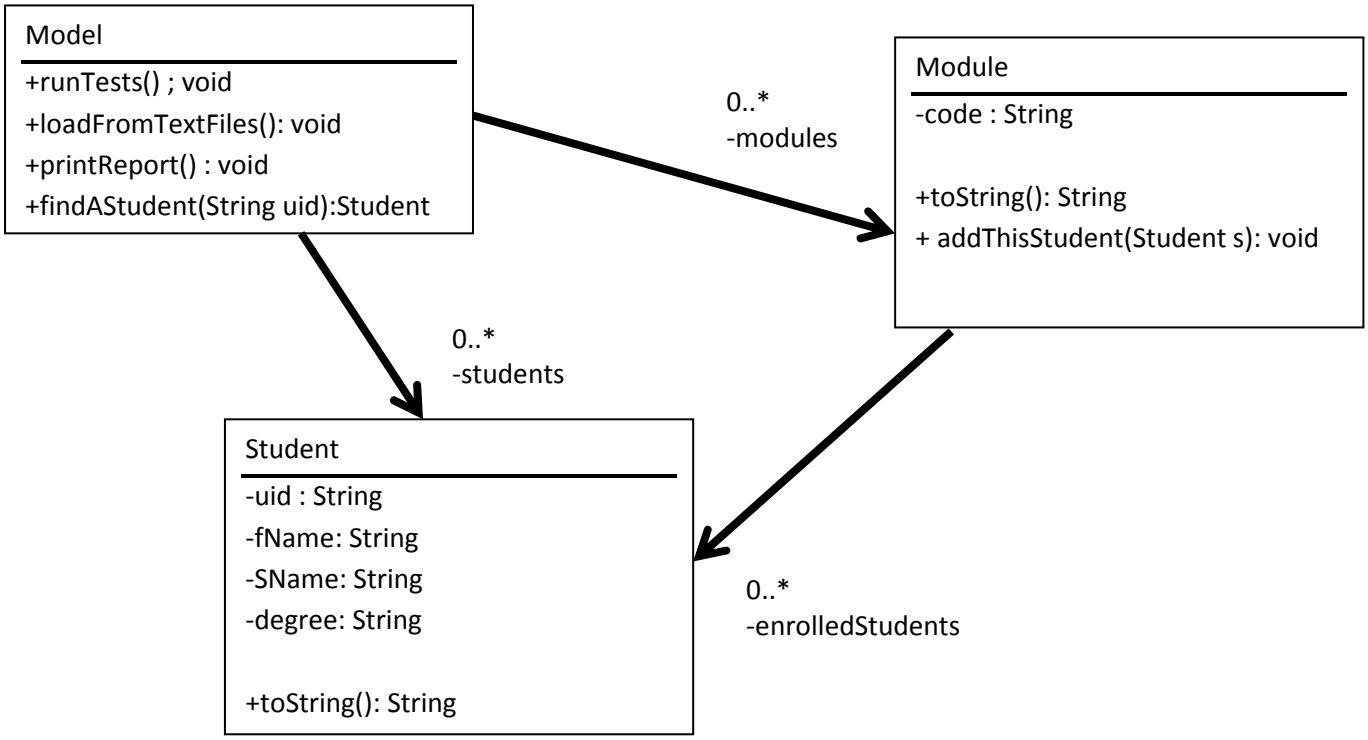
CS12420 2013-2014 - Mini Assignment 1 – Input Output

This mini assignment (worth 10% of your mark) is to put in practice three things:
1. Using Eclipse
2. Input and Output
3. Learning stuff on your own

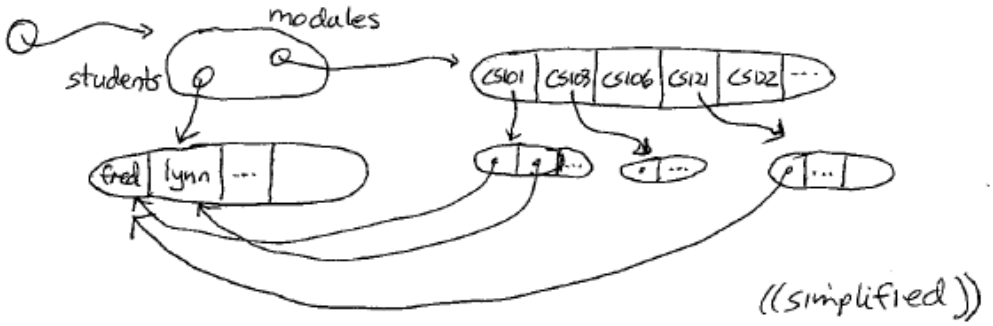
YOUR WORK MUST BE SIGNED OFF BY A DEMONSTRATOR IN YOUR ASSIGNED PRACTICAL SLOT. YOU NEED TO HAVE THIS SHEET SIGNED OFF AT THE LATEST DURING THE WEEK BEGINNING 17th February.

The basic problem is that you have a system that relates students and modules. (Obviously this could be a huge system – we are just looking at a part of it.)

The relationship between them is expressed in this class diagram (note we are omitting the link that allows you to go from Student to their Modules for simplicity):



And at some point in time might be expressed in this object (instance) diagram (using data below).



I would use ArrayLists for this but it is up to you – you can use arrays if you like.

Your task is to:

- Follow the set of instructions that is on BB to open a new project in Eclipse. You may wish to take my code for reading and writing and run it in Eclipse. You will be able to cannibalise it for this assignment.
- Create the classes Model, Student and Module in conformance with the class diagram.
- Write a method for Model that loads the data from 2 text files called students.txt and modules.txt. The data must be of the form:

2 --number of students fyb9 Brown Fred G400 lfr8 Rowlings Lynn G600	5 --number of modules CS10110 2 --number of students in CS10110 fyb9 lfr8 CS10310 0 --number of students in CS10310 CS12130 1 fyb9 CS12230 1 lfr8 CS10610 2 fyb9 lfr8

- In order to model the situation you need to be able to look up a Student and link them into the Module, so your Model must have a method to find students by User ID:

  Student findStudent(String uid) {..}

  And the Module class must have a method:

  void addThisStudent(Student s) {…}

- Your system should read the text files, link up the data and print a report of all the modules and the students who are in them. Something like:

  CS10110
  fyb9 – Fred Brown G400
  lfr8 – Lynn Rowlings G600
  CS10310
  No students
  CS10610
  fyb9 – Fred Brown G400
  lfr8 – Lynn Rowlings G600
  CS12130
  fyb9 – Fred Brown G400
  CS12230
  lfr8 – Lynn Rowlings G600

- Then, your system should also save the data by using Serializable and XML

IN ORDER TO MAKE YOUR LIFE EASIER YOU NEED DO NO ERROR CHECKING (OBVIOUSLY THAT IS VERY UNREALISTIC) AND YOU NEED NO MENU ETC., YOU CAN JUST LET A METHOD DO THE STEPS IN TURN IF YOU LIKE.

So the main might look like this:

```
public static void main (String args[]) {
        Model  theModel=new Model();
        theModel.runTests();
}
```

The steps of runTests() are:

1. Load data from text files, store in the Model and **print report to show base data**
2. Save whole Model using Serializable
3. Change something – for instance you might enrol one of the students on CS10310
4. **Print report to show change**
5. Read Model back from the serialized file
6. **Print report to show that you are back with old data**
7. Save whole Model using XML (don't forget that every class needs to be a bean ie. have all gets and sets and also a default constructor)
8. Change something – for instance you might enrol a *different* one of the students on CS10310
9. **Print report to show change**
10. Read from the XML file
11. **Print report to show that you are back with old data**

This assignment is worth 10% of your mark for CS12420 and will be marked in your practicals by demonstrators using the following scheme:

- **Mastery of compiling and running in Eclipse** **2%**
- **Step 1** **+4%**
- **Steps 2-6** **+2%**
- **Steps 7-11** **+2%**

- **Showing you can read the text file and print out the contents and use Eclipse (but not link the data or do the other bits)** **4% total**

If you need refreshing on any of this look at
http://www.aber.ac.uk/~dcswww/Dept/Teaching/CourseNotes/2013-2014/CS12230/codeExamples/4-5-6-weeks-in-lab/contacts/
SimplerApplication.java has a menu which you can use if you like – but more importantly it has the modelling of many items of data.

CS12230 people – remember that we used a deprecated format for methods in class diagrams:
We used `+String toString()` instead of `+toString(): String`
Now it is time to change.

Finally, self-directed learning (for glory!!)

Investigate the Java methods associated with Stream and String Tokenizer.
Read in a text file of words with punctuation between them and split off the individual words as you go.