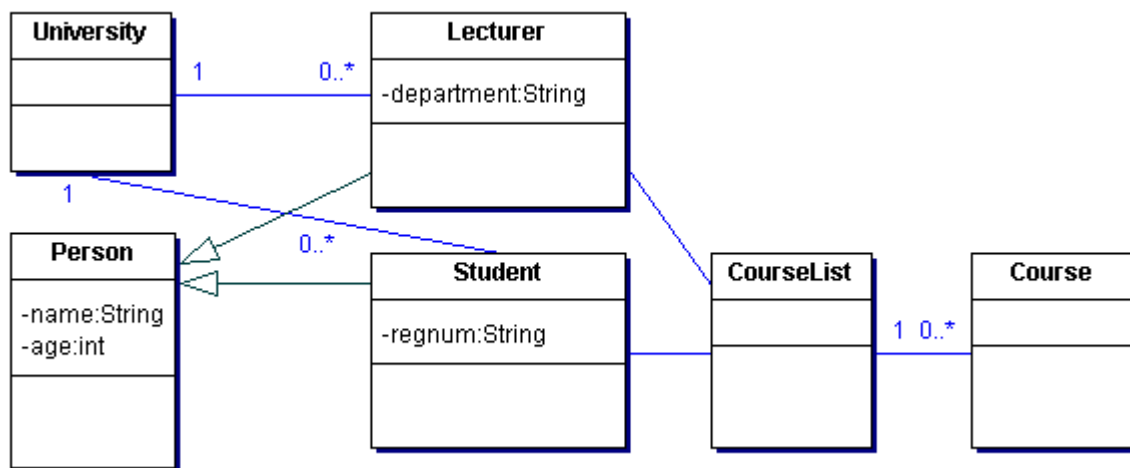


Tutorial 1 (for week starting 24 September)

We have a class `Person` that holds a person's name and age. A `Student` is a `Person` who has a registration number and is taking a set of courses. A `Lecturer` is a `Person` who belongs to a `Department` and is teaching a set of courses. A `University` consists of students and lecturers.

The UML class diagram below is a (partial) first attempt at modelling this University system:



1. Assume that you have a boundary class `UniInterface` through which users can make enquiries about the members of the `University`. Describe a possible chain of messages (operation/method invocations) that would enable an enquiry to determine if a student with a particular registration number is taking a particular course. You will need to make some assumptions about what methods are provided by each class.
2. What is an *abstract class*? Identify a likely candidate for an abstract class in this university system.
3. Suppose that one of the operations offered by class `Person` is `getDetails`. Why will the `Student` and `Lecturer` classes redefine this operation? Using `getDetails` as an example, describe what is meant by *dynamic binding*.
4. Suppose that we implement a collection of `Student` objects as a `Vector` (a standard Java collection class):

```
Vector studentList = new Vector();
studentList.add(new Student(...));
studentList.add(new Student(...));
etc
```

Look up the `Vector` class in the on-line class libraries. Study the description of the method `public boolean contains(Object o)` that can be called like this:

```
Student s = ...;  
if ( studentList.contains(s) ) ...
```

What should we consider adding to the `Student` class so that this works properly if we consider that two `Students` are the same if their registration numbers are the same?

5. What is the output of the following two method invocations? `println` and `+` are each being used in two different ways. Explain the differences between the two statements, and discuss the general case which this is an example of. What is this effect called?

```
System.out.println("The sum of " + x + " and " + y + "is");  
System.out.println(x + y);
```

How should `Student` be modified so that

`System.out.println(someStudent)` ; displays the student's name?