Object Oriented Programming In Python

Activity Sheet: Week 3

Task 1: Find the Classes.

Suggest classes and attributes

Each form has a teachers and a classroom. The pupils are divided between the forms, so that everyone is in exactly one form. The pupil's age at the start of the academic year is recorded

Task 2: Enhance the listRoll method

The method to list the pupils in the form just gives the names of the pupils. It should also:

- Show the teacher and room number, if known, on the first line.
- Show the pupil's age, if known

Task 3 Add a Method to Update the Age

Add a new method to the Form class to provide a way to update the age of a pupil. You should use the findPupil method and other existing methods as far as you can.

Task 4: Add a Teacher Class

The teacher is just known by his/her name. This is not good as really there is other information about teachers. Create a Teacher class, with suitable attributes. Use a teacher object as an attribute of the Form class, rather than just a string.

Task 5: Create a User Interface to Test the Form / Pupil Classes

Create a simple command line interface that can be used add forms and pupils and update information about existing pupils.

Additional Task: Create a More Interesting Scenario

Suggest alternative scenario (not about Forms and Pupils) of similar complexity.

- Write some scenario text
- Write a 'starter' program
- Set some graded tasks to develop the starter program

Homework Task 1: Concepts

The following table shows the concepts we have identified for interaction between classes and the methods. Complete the table to show which concept is introduced in each method.

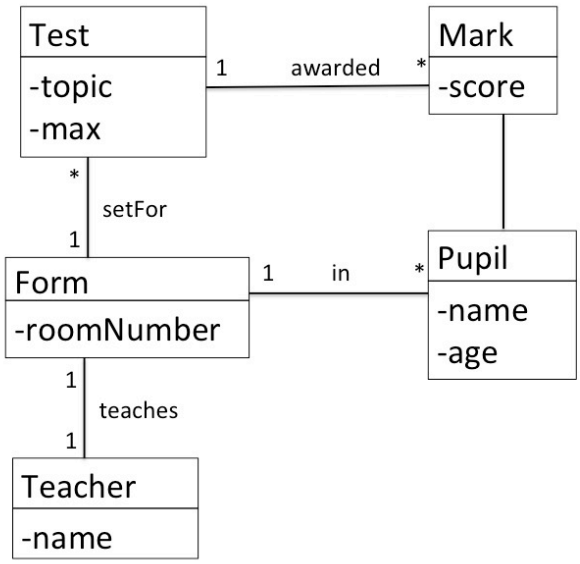
Concept	Present in Method				
	(week 2)	listRoll	AddPupil	AddNewPupil	findPupil
Object as a value (variable, list item, …)					
Object as an attribute value (has-a relationship)					
Object passed as a parameter					

In your view:

- Are these concepts hard to understand? If so, why?
- At these concepts important?
- Are there any more interactions that could be distinguished?

Homework Task 2: No Forms without Tests

This homework asks you to enhance the Form and Pupil example so that we can keep track of the marks in different tests in a form. The following class diagram (without methods) shows the extended Form:



You goal is to implement these classes and create a simple user interface to add tests, pupils and marks for each pupil. To complete the work in stages:

- Code the new classes, initially without methods
- Design the interface dialogue and the methods it will call
- Implement these methods, with additional methods wherever appropriate.