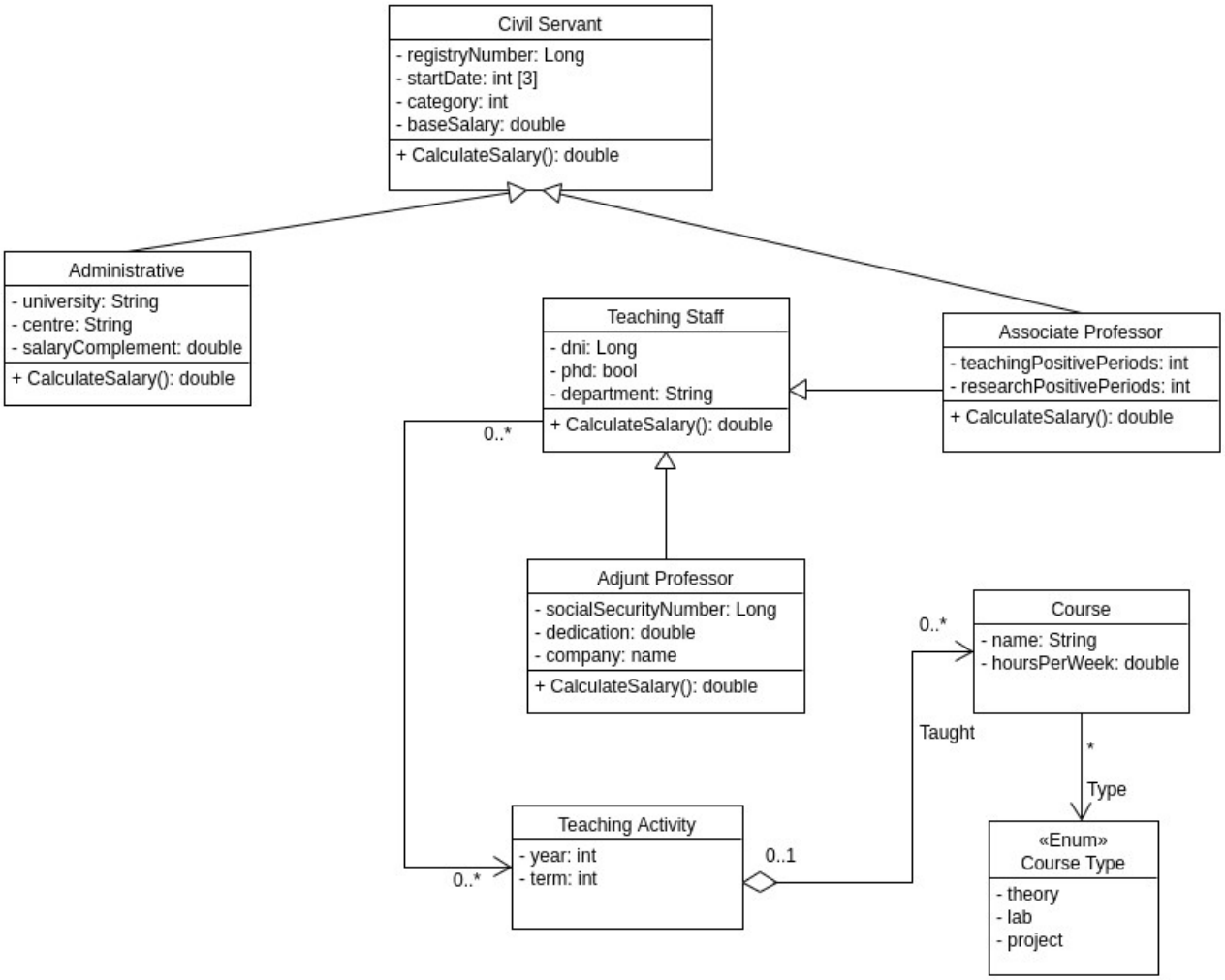
EXERCISE 3

a)



As we can see in the diagram, there exist a class for each type of staff.
First of all we have the Civil Servant class, that stores the required data for a civil servant. The start date could be stored as a Date type, but we have decided to use a buil-in data type to store it as an array of 3 integers. It includes a method, which is CalculateSalary(), that will return the baseSalary attribute.
Then there is a class for the Administrative position, that inherits the attributes and functions of the Civil Servant class. The centre and university could have been designed as two different classes, but we have decided to store them as strings. The function CalculateSalary will call to the parent class to get the base salary and will add to that amount the salaryComplement.
The Teaching Staff class stores the id as a long, the information about the PhD as a boolean variable and the department as a String. This last attribute could have been implemented as another class, but as the design does not require it, we finally decided not to do it. We have considered that a Teaching Staff is not a Civil Servant because it was not written on the assignment, and so this class does not inherit its properties.

The Associate Professor class has a double inheritance, as it is at the same time a civil servant and a teaching staff.
The Adjunt Professor inherits the attributes and functions of the Teaching Staff class and stores its own attributes.
The last three classes rewrite the method CalculateSalary adding to it the salary complements.
Finally there is a class that stores the Teaching Activities for each professor, year and term. The Course class is agregated to this last one and stores the name of the course, the hours per week and an enumeration with the lesson type.

b)

If we want to implement it in Java we will have to change the Associate Professor class because it has multiple inheritance and it is not permited in this programming language. Therefore, we have decided to copy the attributes of the Civil Servant class into the Associate Professor one. It may be not the best solution, as we are repeating twice the same content, but it will solve the problem and it will be possible to implement it Java.