

Lab session 5: Create a Complete Application

24292-Object Oriented Programming

1 Introduction

In this lab session you have to implement the design of Seminar 5 that models a hospital manager. The aim is to fill the gaps in the corresponding classes and methods to allow the project to compile and execute as expected.

2 Hospital Manager

This will be the main class. It should contain the list of hospitals, doctors, administratives and patients as attributes of the class. Also the methods for adding or removing elements from lists.

In the main method, you have to create an instance from *HospitalManager* and call to the methods to add objects in the lists. You should create in a specific order:

1. Add hospitals
2. Add doctors with their specialities
3. Add administratives
4. Add rooms
5. Add beds for each room
6. Assign administratives to hospitals
7. Assign doctors to hospitals where they work
8. Add different residents and visitors to each Hospital
9. Create Visits
10. Assign beds to residents
11. Test everything is assigned properly

After that, you can try to simulate in the HospitalManager, a date that increments by one and doctors that visit patients as scheduled and adding and removing visits and patients from their lists.

Another test is to sort Patients by age or number of visits, and print the sorted list.

3 Design

A possible solution design (with some added features) from the Seminar 5 is in the following picture:

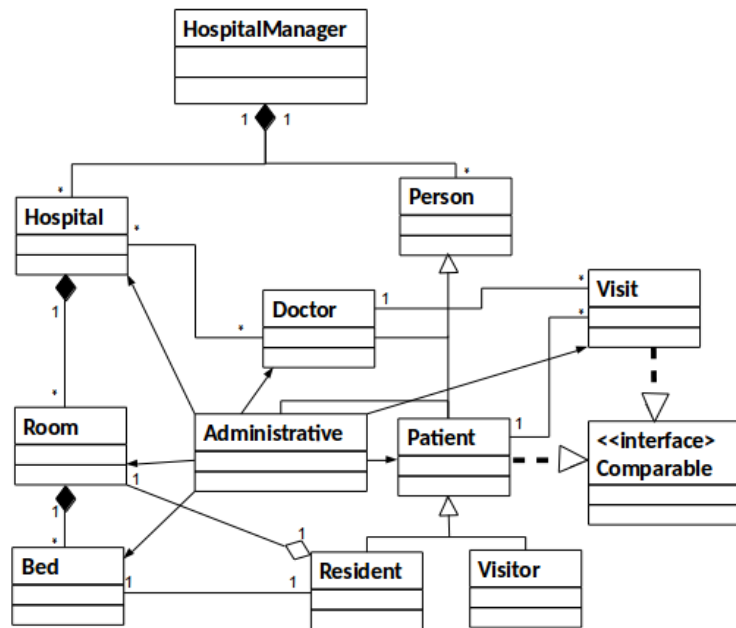


Figure 1: HospitalManager Class Diagram

Even though it is not shown in the diagram, Hospital has a strong aggregation with Visit, but an Administrative is in charge of creating new visits. Although a reference of visit will be send to the corresponding Doctor and Patient.

4 Code

Almost all code is supplied for this Lab session but you should fill the gaps and understand when you need to use inheritance and implementation in classes; what to do inside every method; and get the exact same result as the example provided.

The solution to the provided main should look like this:

Doctor Joline (ID 1) has specialities :
General
Doctor Joline (ID 1) has the following visits :

Doctor Antoine (ID 2) has specialities :
General
Cardiologist
Doctor Antoine (ID 2) has the following visits :

Administrative Clarise (ID 3) has assigned bed to
Resident Jaume (ID 87634, age 19) is assigned to
Room 0 Bed 0 and Doctor Joline (ID 1)
Administrative Clarise (ID 3) has assigned bed to
Resident Monica (ID 34532, age 25) is assigned to
Room 0 Bed 1 and Doctor Antoine (ID 2)
Administrative Clarise (ID 3) has assigned bed to
Resident German (ID 62452, age 50) is assigned to
Room 1 Bed 0 and has no doctor
Administrative Clarise (ID 3) has assigned bed to
Resident Maria (ID 21411, age 37) is assigned to
Room 1 Bed 1 and Doctor Joline (ID 1)
Administrative Clarise (ID 3) has not found bed **for**
Resident Francesc (ID 12999, age 88) and has no
room neither bed and has no doctor

Hospital Sant Joan de Deu

Administratives:

Administrative Clarise (ID 3)

Doctors:

Doctor Joline (ID 1)

Doctor Antoine (ID 2)

Patients:

Resident Jaume (ID 87634, age 19) is assigned to
Room 0 Bed 0 and Doctor Joline (ID 1)
Resident Monica (ID 34532, age 25) is assigned to
Room 0 Bed 1 and Doctor Antoine (ID 2)
Resident German (ID 62452, age 50) is assigned to
Room 1 Bed 0 and has no doctor
Resident Maria (ID 21411, age 37) is assigned to
Room 1 Bed 1 and Doctor Joline (ID 1)
Resident Francesc (ID 12999, age 88) and has no
room neither bed and has no doctor
Visitor Carme (ID 78678, age 63)

Rooms:
Room 0
Bed 0
Bed 1

Room 1
Bed 0
Bed 1

Hospital de Barcelona
Administratives:
Administrative Pere (ID 4)
Doctors:
Doctor Joline (ID 1)
Patients:
Visitor Xavi (ID 12841, age 43)
Visitor Fatima (ID 26256, age 65)
Visitor Johan (ID 62213, age 22)
Visitor Johanna (ID 26268, age 10)
Visitor Jan (ID 99887, age 90)
Rooms:
Room 0
Bed 0
Bed 1

Room 1
Bed 0
Bed 1

5 Submission

As usual, in the main method you should create instances of the various classes, and test the different methods to make sure that they work as expected.

The deadline for this lab session is 30 December at 23:55. Do not forget to write a report explaining your code, ideas, problems and how you solved them.