Hospital Management System (HMS)

Let's put Inheritance into practice!

Since we now have a basic understanding of inheritance and IS-A test, let’s now put them into practice through this simple exercise. In this assignment, we will implement a simple class hierarchy for users of a Hospital Management System (HMS). An HMS can help Hospitals to track everything from user registration to patient treatment to rooms allotted to in-patients and so on. In this exercise, let’s restrict ourselves to implementing only different types of Users.

Users can be of three types: Patient, Doctor, and Nurse.

Doctor & Nurse are Staff members (i.e., Doctor IS-A Staff Nurse IS-A Staff) and all of them (i.e., Patient, Doctor, Nurse, and Staff) are of type User.

Below are details of each class along with information about their instance variables including their data types indicated in parenthesis.

**User**: id (long), firstName (String), lastName (String), gender (String), email (String)

**Patient**: patientId (long), insured (boolean)

**Staff**: staffId (long), yearsOfExperience (int), description (String), salary (double)

**Doctor**: doctorId (long), specialisation (String)

**Nurse**: nurseId (long)

*Note: Ids should be inherited.*

As part of this assignment, you would implement the above classes. Classes should reflect the **inheritance relationship**. Classes should include the getters & setters for each of the fields. Make sure to follow correct naming convention for getters & setters as one of the fields is a boolean (covered in Naming Conventions part of Classes 1 lecture). Notice how the variables have been named (e.g., yearsOfExperience) as discussed in naming conventions lecture.

Finally, make sure to use private access modifiers for all fields and getters setters would be public (remember Information Hiding principle & Encapsulation).

Thanks & Happy Coding!