**Advanced Enterprise Java**

**Project-1**

Submission by Swapnil Patel (1966690)

Table of Contents

[How to Execute: 3](#_Toc127299182)

[Code Explanation: 4](#_Toc127299183)

[Formatting 4](#_Toc127299184)

[Hospital class: 4](#_Toc127299185)

[Person class: 4](#_Toc127299186)

[Patent class: 4](#_Toc127299187)

[Staff class: 4](#_Toc127299188)

[Department class: 5](#_Toc127299189)

[OperationsStaff class: 5](#_Toc127299190)

[Doctor class: 5](#_Toc127299191)

[Surgeon class: 5](#_Toc127299192)

[Receptionist class: 5](#_Toc127299193)

[Technician class: 5](#_Toc127299194)

[Surgical Technologist class: 5](#_Toc127299195)

[Driver class: 5](#_Toc127299196)

[Output: 6](#_Toc127299197)

# How to Execute:

To execute just run the Driver.java file inside src folder.

> javac Driver.java && java Driver

# Code Explanation:

## Formatting

For formatting the output each class that which gets an object are assigned with a level. This level decides the no. of spaces that each object print after so we get indented look in the output.

Text

Description automatically generated

## Hospital class:

We have Hospital class that will have various attributes along with a constructor and a total department and patient limit counter: Since the number of department and patient is dynamic, we are using List collection to store them.

We can add department to the Hospital but after the limit has exceeded add method will through a custom exception.

Same is the case for patients.

## Person class:

This class defines constructor to basic properties of a person also translates string to date object using a method string2Date(). It has an abstract method called getBirthdate as birth date year of Staff should be hidden. But for patients it should be shown

It has a toString overloaded method to print all attributes of a person.

## Patent class:

Its constructor calculates the age of the patient along with the date formatting is printed here along with the year of birth for checkup.

It has a toString overloaded method to print all attributes of a patient.

## Staff class:

Staff class extends the Person class and add job specific documentation attributes like language, certificates etc.

It redefines the getBirthDate method hence making this class concrete.

## Department class:

Department class has a name to it and various staff. It has a list of Staff superclass and depending on the type of class. It also has an upper limit on the number of staff members it can accommodate.

## OperationsStaff class:

This class is intermittent class which can be further subclassified into Doctor and Nurse class and it extends staff class so has all it’s attributes.

## Doctor class:

It extends the Operation Staff class and has all its attributes plus the 2 additional array of parameters specialty and location which can be 5 each.

## Surgeon class:

It extends the doctor class and will have a specific attribute of a surgeon.

## Receptionist class:

This class against is a sub-sub class of Staff and has a different classification of administrative staff.

## Technician class:

This is a sub class of Technical Staff which is associated with technical tasks of a hospital.

## Surgical Technologist class:

This class is a sub-sub class of Staff and has surgical technical specialty.

## Driver class:

We create a Hospital Object with limit set on the number of departments.

Then we keep on adding the departments to the Hospital under a try catch as it can throw custom exception if added departments exceed total departments.

We created an array of dummy dates to be used for creating data for Staff and patients.

Now Now with polymorphism we create Person reference and assign Patient object to it. Unde try catch as after the limit has exceeded add method will through a custom exception.

Now with polymorphism we create Staff reference and assign Nurse, Surgeon, Receptionist Technician and SurgicalTechnologiest object to it but after the limit has exceeded add method will through a custom exception.

# Output:

Text

Description automatically generated