



Object-Oriented System Design

Programming Assignment #2

Hospital Management System

- Assignment #1을 기반으로 inheritance와 polymorphism 기능을 이용하여 병원관리 시스템(hospital management system)을 확장 구현한다.
- Patient를 입원환자(inpatient)와 외래환자(outpatient)로 구분하여 진료 및 치료를 받을 수 있도록 한다.
- 의료기관 종사자(hospital personnel)를 doctor와 nurse로 세분화하여 시스템을 사용할 수 있도록 한다.

Hospital Management System

- Assignment #2에 다음과 같은 class를 포함한다
 - Patient
 - InPatient
 - OutPatient
 - HospitalPersonnel
 - Nurse
 - Doctor
 - HospitalManagementSystem

Class Patient

- private int pid
- private String name
- private int age
- private Doctor doctor // 지정 의사
- private String prescription

- Patient(int pid, String name, int age)
- Patient(int pid, String name, int age, Doctor doctor)
- setDoctor(Doctor doctor)
- setPrescription(String prescription)
- String toString() // 환자 정보를 String 형식으로 리턴
- outputPatientInfo() // toString()을 call하여 환자 정보를 출력
// Polymorphism이 작동하여 Patient class의 세부
타입에 따라 출력되는 내용이 달라진다.

Class InPatient

InPatient extends Patient

- Date inDate //입원일
- Date outDate //퇴원(예정)일
- InPatient(Date in, Date out)
- setInOutDate(Date in, Date out)
- String toString() //Override하여 입퇴원 정보를 포함한 정보를 리턴

Class OutPatient

OutPatient extends Patient

- Date appointment //진료 예약 일시
- OutPatient(Date aptment)
- String toString() // Override하여 진료 예약 일시를 포함하여 리턴
- setAppotintment(Date date)

Class HospitalPersonnel

public abstract class HospitalPersonnel

- int uid
- String password
- String name
- HospitalManagementSystem hospitalsystem
- HospitalPersonnel(int id, String name, String password, HospitalManagementSystem hospitalsystem)
- abstract void managePatient(); // Polymorphism 작동
- abstract void updatePatient();

Class Nurse

Nurse extends HospitalPersonnel

- Nurse()
- managePatient() // Override – 간호사 메뉴 화면 수행
- updatePatient(int pid, Date start, Date end) // 환자 입퇴원일, 진료예약일 변경. 진료 예약일의 경우 end는 null 값을 갖는다.
- registerPatient(Patient patient, Doctor doctor) // 환자 정보를 담당 의사와 함께 저장한다

Class Nurse

- managePatient() // Polymorphism이 적용됨
 // 아래와 같은 nurse의 main 화면을 출력하고 키보드 입력에 따라 동작한다
 // register 메뉴는 환자를 시스템에 추가한다
 // update 메뉴는 환자의 입퇴원일이나 진료예약일을 저장한다
 // 환자 정보 출력은 특정 환자나 전 환자 정보를 출력할 수 있어야 한다.

 ===== Welcome to Hospital Management System (Nurse)=====
 1 - Register patient
 2 - Update patient information
 3 - Output patient information
 0 - Exit

Class Doctor

Doctor extends HospitalPersonnel

- Doctor()
- managePatient() //Override – doctor 메뉴 화면 수행
- updatePatient(int pid, String prescription) // 처방전을 입력한다

Class Doctor

- managePatient() // Polymorphism이 적용됨
 // Doctor의 main 화면을 출력하고 키보드 입력에 따라 동작한다
 // update 메뉴는 의사의 처방을 환자 정보에 저장한다
 // 환자 정보 출력은 특정 환자나 전 환자 정보를 출력할 수 있어야 한다.

=====Welcome to Hospital Management System (Doctor)=====

- 1 - Update patient information
- 2 - Output patient information
- 0 - Exit

Class HospitalManagementSystem

- HospitalPersonnel[] hospitalPersonnelList // 의사 + 간호사
- Patient[] patientList // 등록 환자

- HospitalManagementSystem(HospitalPersonnel[] hp, Patient[] pa)
- addDoctor(Doctor doctor)
- addNurse(Nurse nurse)
- addPatient(Patient patient)
- outputPatient(int pid) // 특정 환자 정보 출력
- outputPatient() // 모든 환자 정보 출력

- login(String name, String password)
 - Name과 password를 입력받은 후 managePatient()를 호출한다.
 - 그 결과 polymorphism에 의해 사용자가 nurse인지 doctor인지에 따라 해당하는 managePatient 주메뉴 화면이 표시된다

Class HospitalManagementSystem

- public static void main(String[] args)

// 프로그램의 main method로 아래와 같이 의사, 간호사, 환자 정보를 추가하는 초기화를 수행한다. 명시되지 않은 instance variable의 값은 적절히 선택한다.

- Doctor 1 - name: Emma; password:123456;
- Doctor 2 - name: Olivia; password:123456;
- Nurse 1 - name: Ava; password: 123456;
- Nurse 2 - name: Mia; password: 123456;

- Patient 1 - name: Bob; age: 25; 담당의사 미정 (base class인 Patient class의 object)
- Patient 2 - name: Tom; age: 23; 담당의사 Emma (inpatient로 입퇴원 기간 설정)
- Patient 3 - name: Alice; age: 18; 담당의사 Olivia (outpatient로 진료일을 적절하게 지정)

Submission

- LMS 과제 항목에 작성한 소스코드 패키지(Assignment2_code_이름_학번)와 설계 문서(Assignment2_이름_학번)를 압축하여 제출
- 설계 문서
 - 설계 문서에서는 프로그램 구조 설계와 주요 method 설명한다. 또한 프로그램의 실행 방법을 설명하며 문서 작성 후 pdf 로 변환하여 제출
- 제출 기한
 - 5월 24일 자정 (23:59 PM)
- 지연 제출 페널티
 - 마감 시한 이후 24시간까지 50%, 48시간 이후 75%, 72시간 이후에는 100% 감점