

소프트웨어 분석 설계



# 학과 : 컴퓨터공학과

학번 : 20170677, 20180658

이름 : 오융택, 이정탁

과목 : 소프트웨어 분석 설계 교수님 : 윤회진 교수님 제출일 : 23.4.19



- 1 -

연습문제 2번

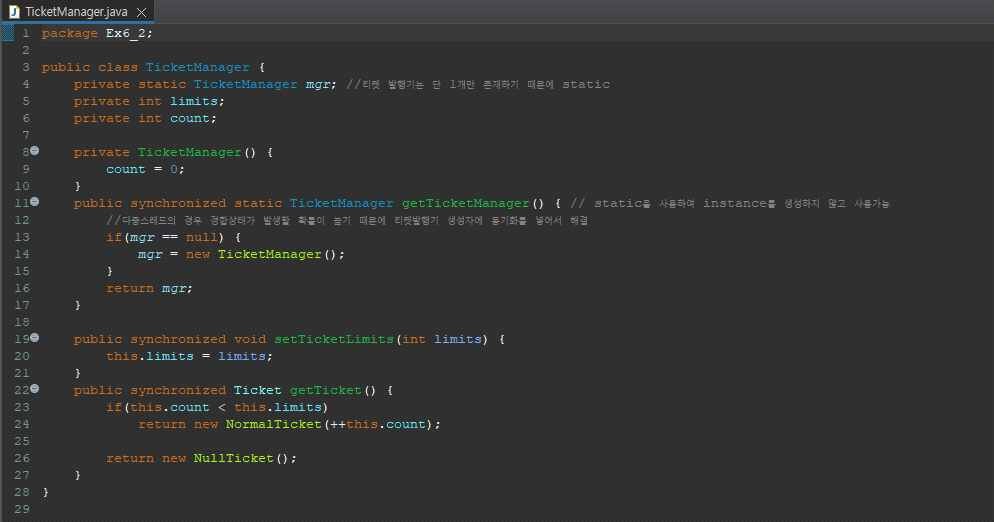


Q(1) 요구사항 5가지를 반영하는 클래스들이 P.430 표에 정리되어져 있다. 이 가운데 singleton 패턴을 적용해야 하는 클래스는 무엇인가?

A(1) TicketManager 클래스에서 singleton패턴을 적용해야 한다.

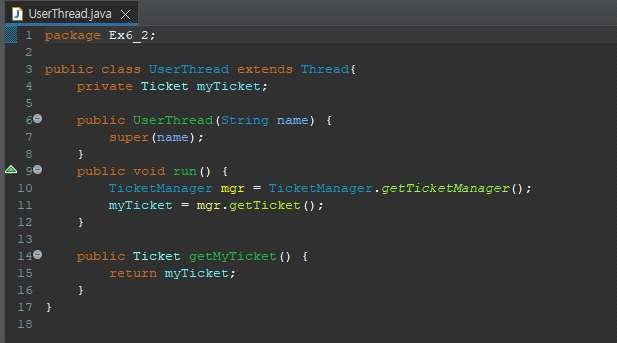
Q(2) 정답 코드 (p.430)를 작성 : thread의 수가 15개 ticket의 수가 5개 일 때 거의 순서 변 화없이 일정하다… Thread의 수와 Ticket의 수를 늘려줄수록 다양한 순서들이 등장한다.

A(2)thread의 수를 30개, ticket의 수를 15개로 설정함

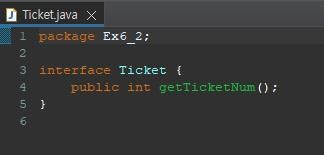


TicketManager.java

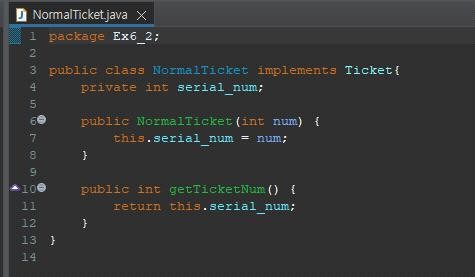
- 2 -



UserThread.java

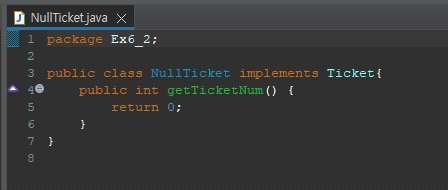


Ticket.java



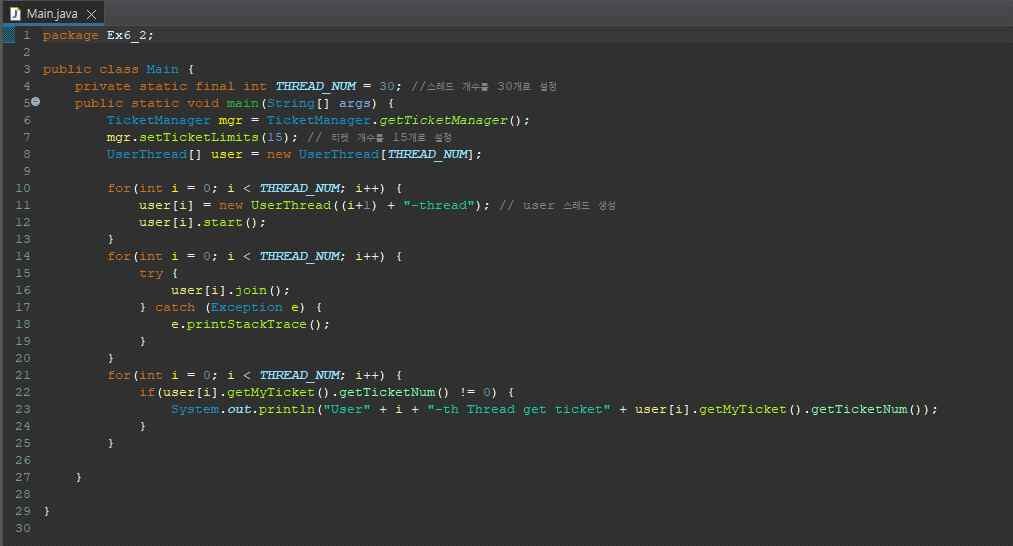
NormalTicket.java

- 3 -



NullTicket.java

- 4 -



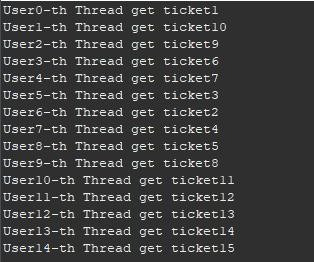
Main.java

Q(3) 코드에서 Singleton 패턴을 설명할 수 있어야 한다.

A(3) singleton에서는 인스턴스가 오직 하나만 생성되는 것을 보장해야 한다. 티켓 발행기가 1대인데 여러 개의 인스턴스가 생성되게 되면 조건에서 벗어난다.

private static TicketManager *mgr*; // 조건에서 티켓 발행기는 단 1개만 존재해야 하기 때문에 static 사용

public synchronized static TicketManager getTicketManager() { // static을 사용하여 instance를 생성하지 않고 사용가능, 다중 스레드의 경우 경합상태가 발생할 확률이 높기때문에 티켓 발행기 생성자에 synchronized를 넣어서 해결



실행 결과

따라서 실행 결과가 ticket의 시리얼 넘버가 겹치지 않게 발행되어진다.

- 5 -