

DB 프로젝트 1-3 보고서

2015-11923 최한결

본 프로젝트에서는 직접 DB를 설계하고 그 DB를 이용해 공연 티켓 예매를 시뮬레이션 하는 간단한 응용 프로그램을 만들어보았다.

개발 환경

OS: MacOS catalina

IDE: Pycharm

Python version: python 3.8

핵심 모듈과 알고리즘에 대한 설명

우선 DB schema를 구현할 때 audience, performance, building은 프로젝트 스펙 문서와 동일하게 두었고 performance에 building의 ID를 building_id라는 이름으로 foreign key로 설정해 performance와 building이 many to one의 관계를 가지게 했다. 또 예약 정보를 seat이라는 table을 두어 관리하게 했는데 seat은 performance_id와 audience_id를 foreign key로 가지고 seat_number와 performance_id를 primary로 가지는 table로 설정했다. command 10을 이용해 performance를 building에 assign할 때 해당 performance_id와 해당 building의 capacity만큼의 seat_number를 가지는 seat record를 만들고 audience가 booking시에 해당 performance_id, seat_number에 해당되는 record의 audience_id를 update하는 식으로 구현했다.

performance의 building_id와 seat의 performance_id 모두 on delete cascade 옵션을 설정해두어 해당 building 혹은 performance가 삭제될 때 같이 삭제되도록 만들어 스펙을 만족시켰고 seat의 audience_id는 on delete set null 옵션을 걸어주어 해당 유저 삭제 시에 예약된 좌석을 null로 바꿔 integrity를 만족시켰다.

구현 내용

스펙 문서에 나온 모든 내용을 구현하였다.

가정한 것들

1. Command 입력 시에 숫자만 들어온다고 가정했다. 이외에도
2. Database reset y/n 입력시에 다른 input이 들어올 경우 Invalid Input이라는 메시지를 출력하게 하였고 reset이 완료될 경우 done! 이라는 메시지가 출력되게 하였다.
3. 가격 출력시에 message.pdf에 출력 방식과 2020_2의 출력 방식이 다르게 표현되어 있어 message.pdf 방식을 따랐다.

■ 가격에 쉼표를 없애고 Total ticket price is 100000 와 같이 표현

컴파일과 실행 방법

main.py가 들어있는 폴더에서 python main.py를 실행하면 된다. pymysql이 설치되어있고 python 버전이 3.8일경우 정상 실행된다.

느낀 점

Db schema를 구성하는 데에 있어 많은 시나리오를 생각해야해서 DB 설계가 어렵다고 느꼈다. 배운 내용을 활용해 DB를 설계하고 그것을 이용해 실제 응용 프로그램을 만들어 보는 것이 재밌는 경험이었다/