

08) Django - Seed Data

1. 목표

- 데이터를 생성, 조회, 삭제, 수정할 수 있는 Web Application 제작
- Seed Data를 활용한 DB 설계
- Django Form을 통해 입력받은 데이터 유효성 검증

2. 준비 사항

1. (필수) Python Web Framework

- Django 2.1.x
- Python 3.6.x

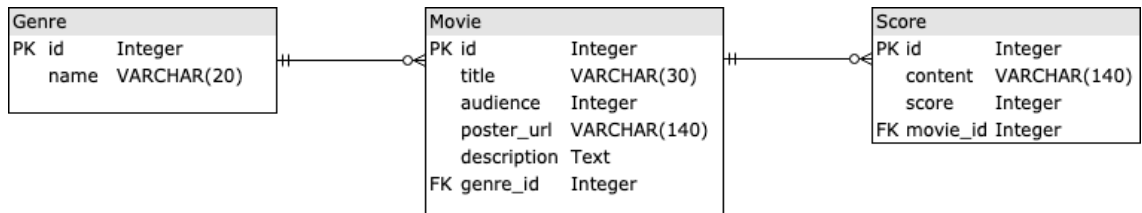
2. (선택) 샘플 영화 정보

- [링크](#)
 - 지난주차 수업에서 활용하였던 코드를 이어서 하셔도 됩니다.

3. 요구 사항

1. 데이터베이스 설계

- `db.sqlite3` 에서 테이블 간의 관계는 아래와 같습니다.



- **Genre**

필드명	자료형	설명
id	Integer	Primary Key
name	String	장르 구분

- **Movie**

필드명	자료형	설명
id	Interger	Primary Key
title	String	영화명
audience	Integer	누적 관객수
poster_url	String	포스터 이미지 URL
description	Text	영화 소개
genre_id	Integer	Genre의 Primary Key(id 값)

◦ Score

Name	자료형	설명
id	Integer	Primary Key
content	String	한줄평(평가 내용)
score	Integer	평점
movie_id	Integer	Movie의 Primary Key(id 값)

2. Seed Data 반영

1. 주어진 `movie.json` 과 `genre.json` 을 `movies/fixtures/` 디렉토리로 옮깁니다.
2. 아래의 명령어를 통해 반영합니다.

```
$ python manage.py loaddata genre.json
Installed 11 object(s) from 1 fixture(s)
$ python manage.py loaddata movie.json
Installed 10 object(s) from 1 fixture(s)
```

3. `admin.py` 에 `Genre` 와 `Movie` 클래스를 등록한 후, `/admin` 을 통해 실제로 데이터베이스에 반영되었는지 확인해봅시다.

3. `movies` App

Genre는 CRUD를 만들지 않습니다.

1. 영화 생성을 위한 사용자 Form

1. (필수) 영화목록 에서 `a` 태그 혹은 버튼을 만듭니다. (Form을 요청하는 URL은 `GET /movies/new` 입니다.)
2. (필수) 템플릿 파일 이름은 `form.html` 으로 영화 수정 페이지와 같이 공유합니다.
3. (필수) `id` 를 제외한 모든 필드에 맞는 `form` 과 `input` 을 제공합니다.
4. (선택) `poster_url` 필드 대신 `poster_image`를 통해 미디어 파일 업로드 기능을 이용하여도 됩니다.

2. 영화 생성

1. (필수) 해당 요청을 처리하는 URL은 `POST /movies/new` 입니다.
 2. (필수) 검증을 통해 유효한 경우 데이터베이스에 저장을 하며, 아닌 경우 오류 메시지와 함께 `form.html` 을 반환합니다.
 3. (필수) 데이터베이스에 저장되면, 영화 정보 조회 `GET /movies/<int:movie_pk>` 로 redirect 합니다.
3. 영화 수정을 위한 사용자 Form
1. (필수) 영화 정보 조회 페이지에서 `a` 태그 혹은 버튼을 만듭니다. (Form을 요청하는 URL은 `GET /movies/<int:movie_pk>/edit` 입니다.)
 2. (필수) 템플릿 파일 이름은 `form.html` 으로 영화 생성 페이지와 같이 공유합니다.
 3. (필수) `id` 를 제외한 모든 필드에 맞는 `form` 과 `input` 을 제공합니다.
 4. (선택) `poster_url` 필드 대신 `poster_image` 를 통해 미디어 파일 업로드 기능을 이용하여도 됩니다.
4. 영화 수정
1. (필수) 해당 요청을 처리하는 URL은 `POST /movies/<int:movie_pk>/edit` 입니다.
 2. (필수) 검증을 통해 유효한 경우 데이터베이스에 변경내용을 저장 하며, 아닌 경우 오류 메시지와 함께 `form.html` 을 반환합니다.
 3. (필수) 데이터베이스에 수정되면, 영화 정보 조회 `GET /movies/<int:movie_pk>` 로 redirect 합니다.
5. (선택) 평점 생성
1. 영화 정보 조회 페이지에서 `form` 을 통해 평점을 작성할 수 있습니다.
 2. 평점 생성 URL은 `POST /movies/1/scores/new`, `POST /movies/2/scores/new` 등이며, 동적으로 할당되는 부분이 존재합니다. 동적으로 할당되는 부분에는 데이터베이스에 저장된 영화 정보의 Primary Key가 들어갑니다.
 3. 검증을 통해 유효한 경우 데이터베이스에 저장을 하며, 아닌 경우 영화 정보 조회 페이지로 Redirect 합니다.
 4. 데이터베이스에 저장되면, 해당하는 영화의 영화 정보 조회 페이지로 Redirect 합니다.

4. 결과 예시

Python Web Framework를 활용해 작성한 모든 파일을 프로젝트 디렉토리에 위치하도록 합니다.

결과물은 반드시 [README.md](#) 으로 활용 하였던 내용을 정리 해주세요.

```
project명/  
...  
...  
...  
README.md
```