DB 설계를 활용한 REST API 설계

1. 목표

- DRF(Django Rest Framework)를 활용한 API Server 제작
- Database 1:N, M:N에 대한 이해

2. 준비사항

A. 개발도구 및 라이브러리

i. Visual Studio Code

ii. Google Chrome Browser

iii. Django 3.2+

iv. Postman

B. Load fixture data

i. 주어진 fixture data를 load하고 프로젝트를 시작합니다.

```
$ python manage.py migrate

# json 파일은 movies/fixtures/movies/ 에 위치합니다.
$ python manage.py loaddata movies/actors.json movies/movies.json movies/reviews.json
```

3. 요구사항

django 프로젝트 이름은 pjt08, 앱 이름은 movies로 지정합니다.
모델 간의 관계 설정 후 다음과 같은 기능을 구현합니다.

A. Actor

i. 배우 데이터 조회

B. Movie

i. 영화 데이터 조회

C. Review

i. 리뷰 데이터 조회 / 생성 / 수정 / 삭제

- 데이터 조회는 JSON 데이터 타입을 따릅니다.
  아래 기술된 사항들은 필수적으로 구현해야 하는 내용입니다.

4. Model

A. 모델 클래스 Actor는 다음과 같은 정보를 저장합니다.

필드명	데이터 유형	역할
name	varchar(100)	배우 이름

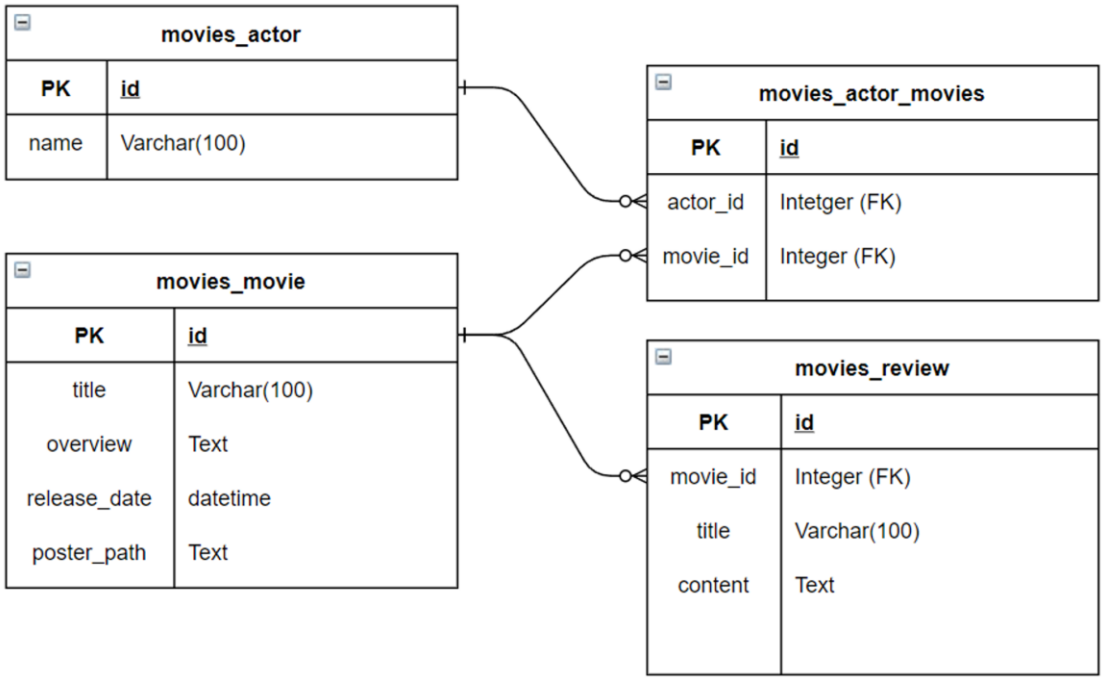
B. 모델 클래스 Movie는 다음과 같은 정보를 저장합니다.

필드명	데이터 유형	역할
title	varchar(100)	영화 제목
overview	text	줄거리
release_date	datetime	개봉일
poster_path	text	포스터 주소

C. 모델 클래스 Review는 다음과 같은 정보를 저장합니다.

필드명	데이터 유형	역할
title	varchar(100)	리뷰 제목
content	text	리뷰 내용
movie_id	integer	외래 키(Movie 클래스 참조)

D. ERD



5. **Admin**

   정의한 모델 클래스 Actor, Movie, Review를 Admin site에 등록합니다.
   Admin site에서 **배우, 영화 데이터를 생성**합니다.

6. **URL**

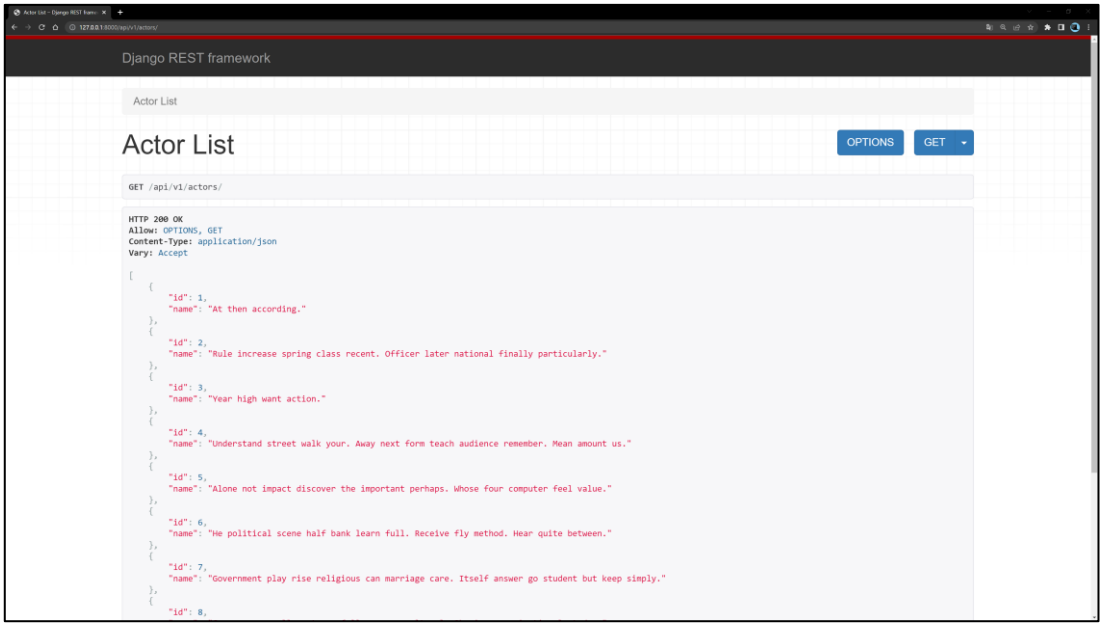
   TMDB(https://developers.themoviedb.org/3/) 문서를 참고하여 RESTful한 URL을 자유롭게 구성합니다.

7. **View**

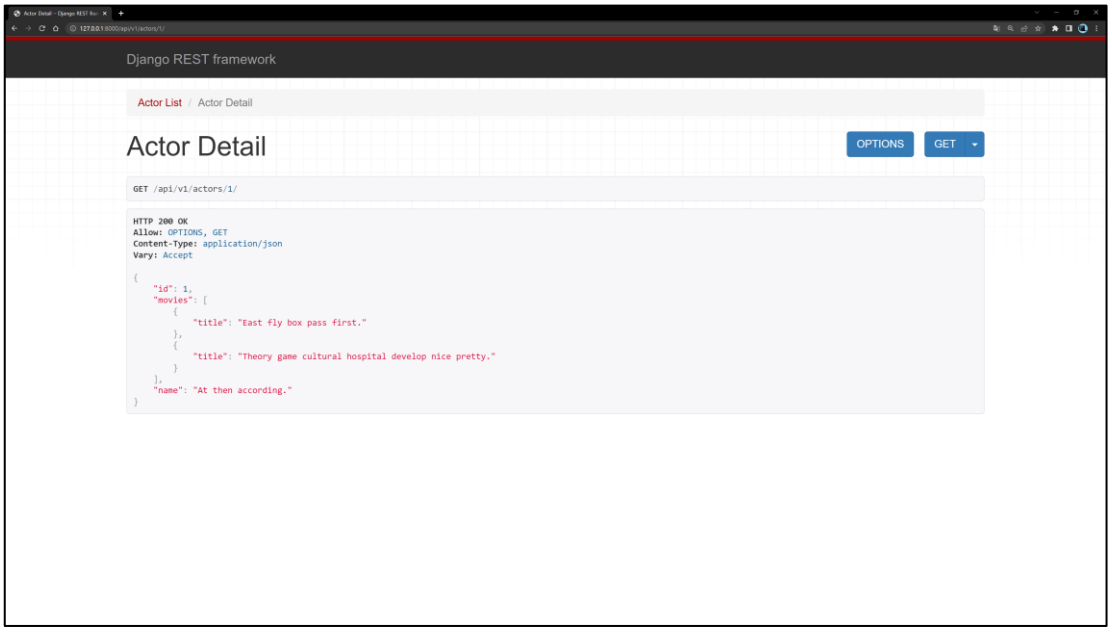
View 함수명	역할	허용 HTTP Method
actor_list	전체 배우 목록 제공	GET
actor_detail	단일 배우 정보 제공(출연 영화 포함)	GET
movie_list	전체 영화 목록 제공	GET
movie_detail	단일 영화 정보 제공(출연 배우와 리뷰 목록 포함)	GET
review_list	전체 리뷰 목록 제공(작성된 영화 포함)	GET
review_detail	단일 리뷰 조회 & 수정 & 삭제	GET / PUT / DELETE
create_review	리뷰 생성	POST

8. 응답 예시

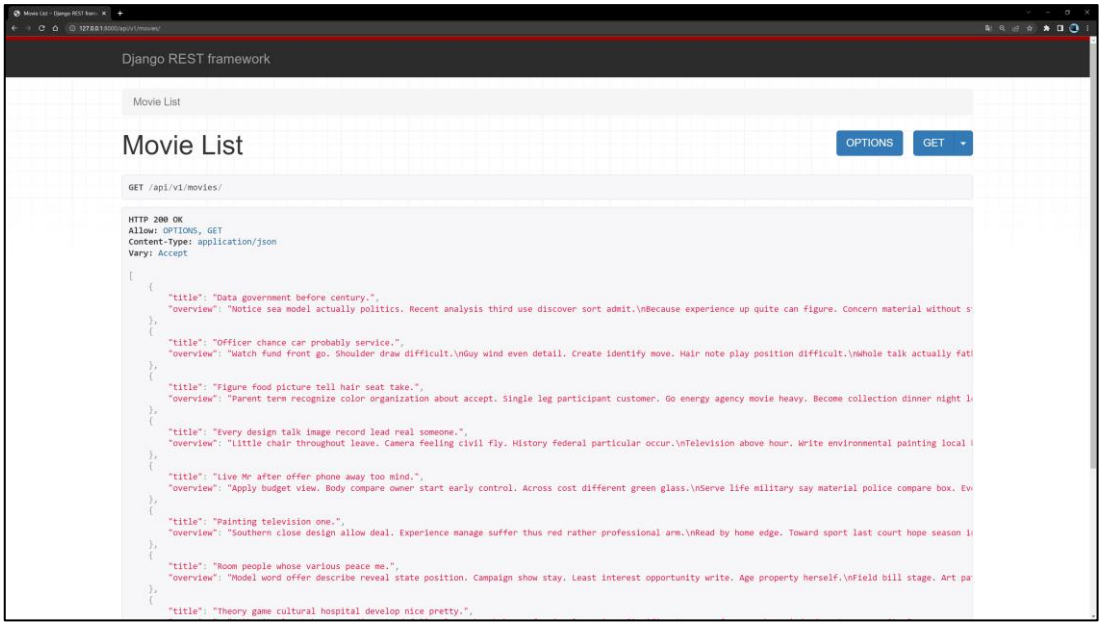
A. GET api/v1/actors/



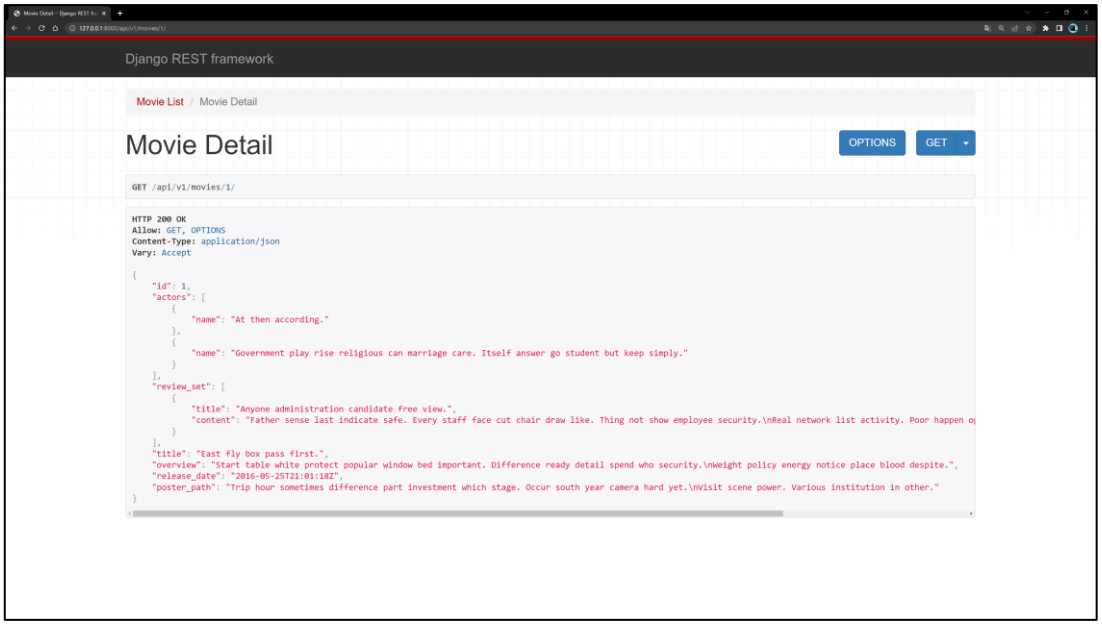
B. GET api/v1/actors/1/



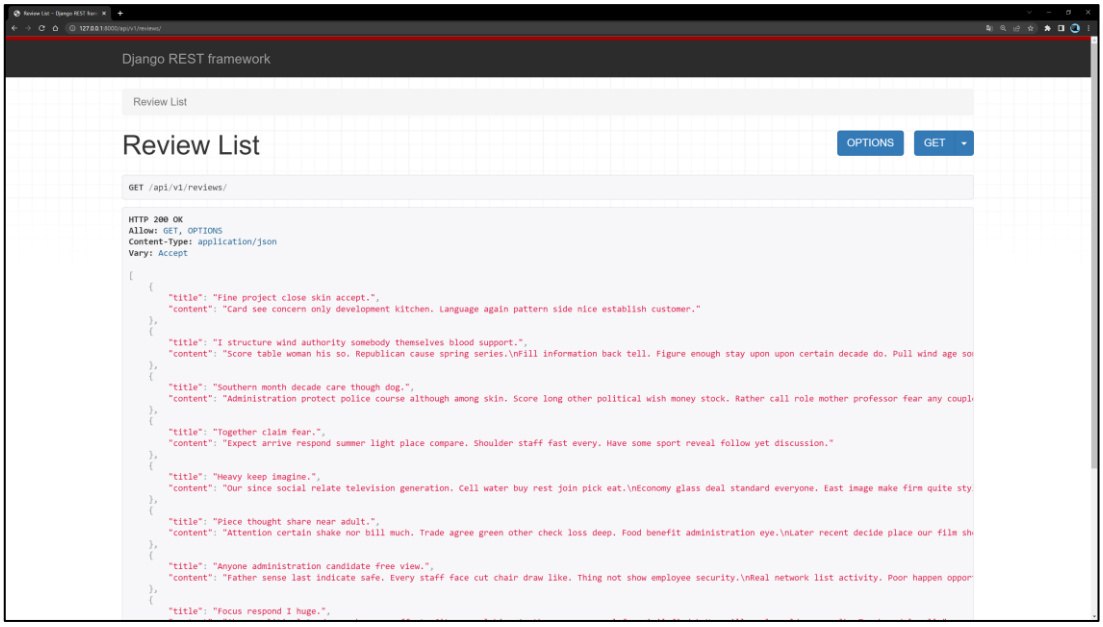
C. GET api/v1/movies/



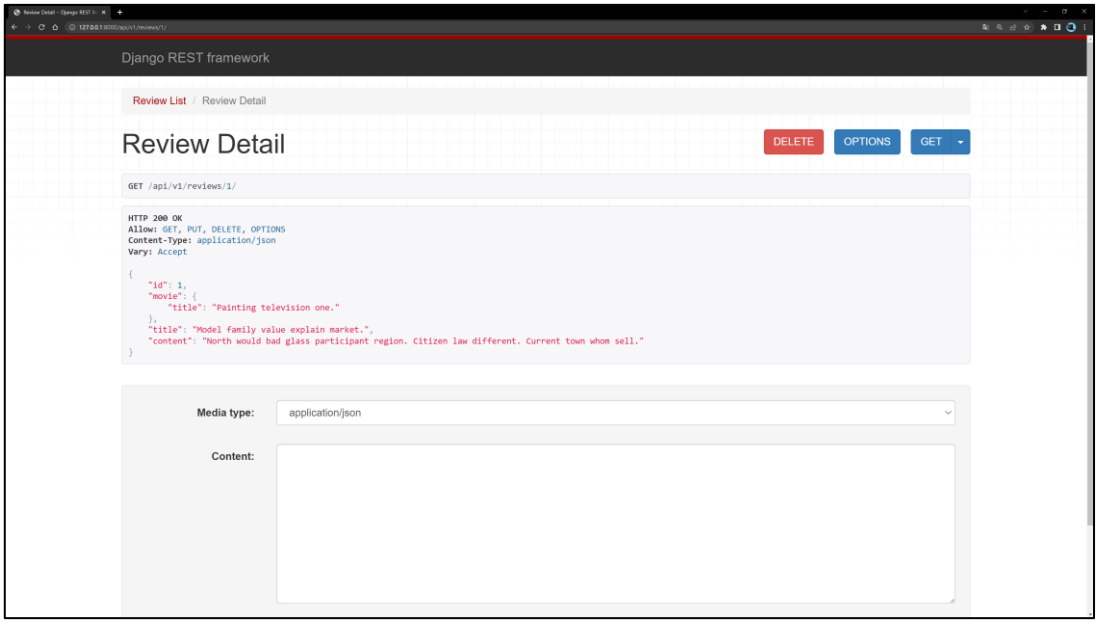
D. GET api/v1/movies/1/



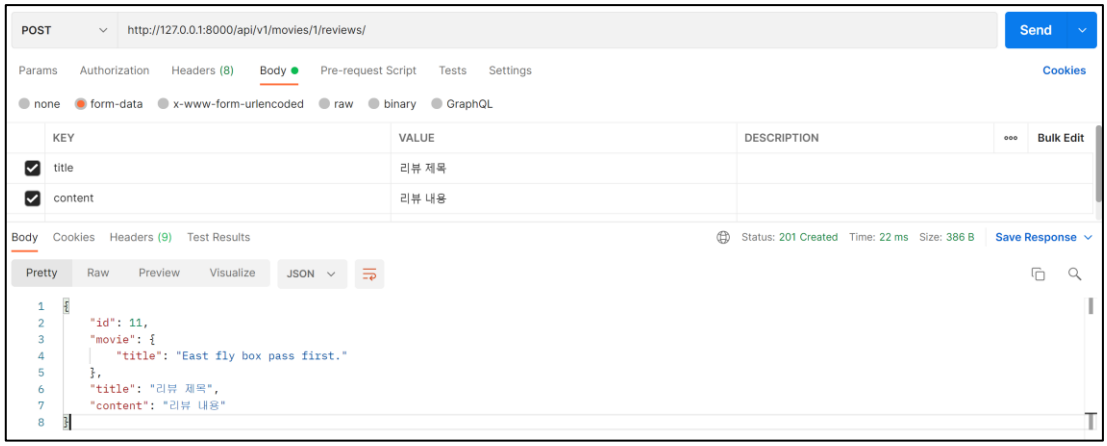
E. GET api/v1/reviews/



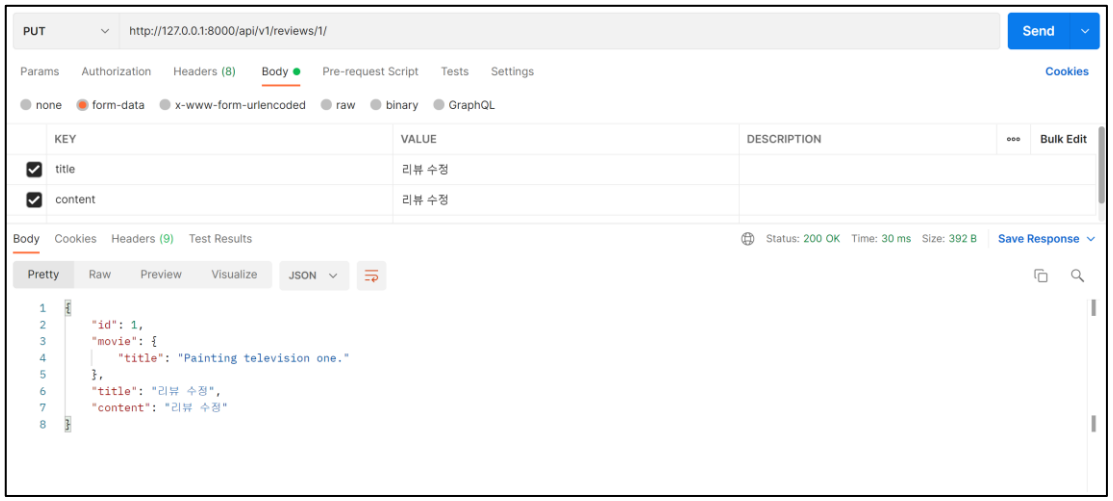
F. GET api/v1/reviews/1/



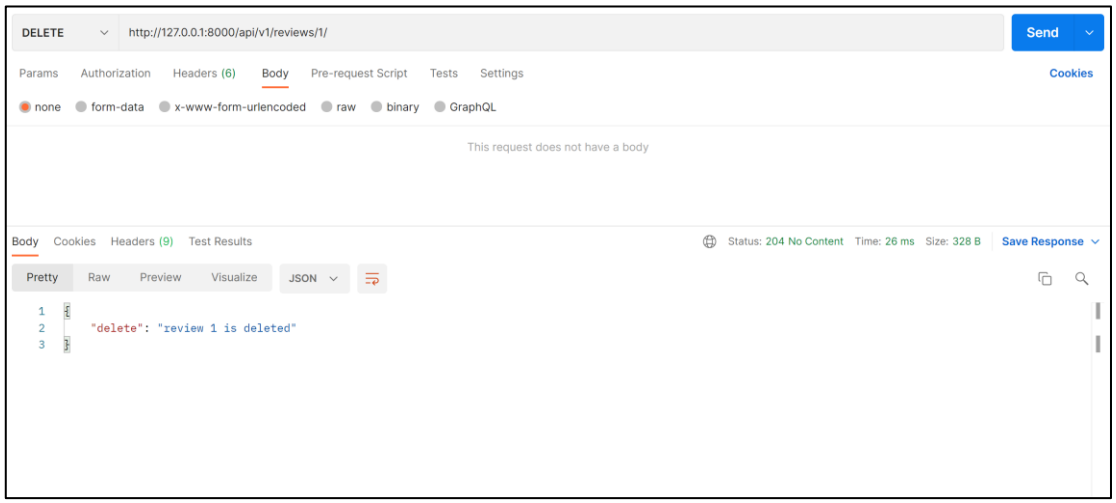
G. POST api/v1/movies/1/reviews/



H. PUT api/v1/reviews/1/



I. DELETE api/v1/reviews/1/



9. 기타

A. 명세에 작성된 것 이외의 아이디어는 자유롭게 구현합니다.

10. 제출

A. 제출기한은 금일 18:00까지입니다. 제출기한을 지켜 주시기 바랍니다.

B. 반드시 README.md 파일에 단계별로 구현 과정 중 학습한 내용, 어려웠던 부분, 새로 배운 것들 및 느낀 점 등을 상세히 기록하여 제출해야 합니다.

C. 위에 명시된 사항은 최소 조건이며, 추가 개발을 자유롭게 진행할 수 있습니다.

D. https://lab.ssafy.com에 pjt08 프로젝트를 생성하고 업로드하여 제출합니다.

E. 반드시 각 반 담당 교수님을 Maintainer로 설정해야 합니다.