

04 PJT

프레임워크 기반 웹 페이지 구현

INDEX

- 프레임워크 기반 웹 페이지 구현
 - 목표
 - 준비사항
 - 요구사항
 - 제출

목표

프로젝트 목표

- 데이터를 생성, 조회, 수정, 삭제할 수 있는 Web application 제작
- Django web framework를 사용한 데이터 처리
- Django Model과 ORM에 대한 이해
- 관리자 페이지를 통한 데이터 관리

준비사항

| 개발도구 및 제공사항

- 개발도구
 - Visual Studio Code
 - Google Chrome
 - Django 3.2+

요구사항

| 공통 요구사항

- 커뮤니티 웹 서비스의 데이터 구성 단계입니다.
- 영화 데이터의 생성, 조회, 수정, 삭제가 가능한 애플리케이션을 완성합니다.
- Django 프로젝트의 이름은 **mypjt**, 앱 이름은 **movies**로 지정합니다.
- 명시된 요구사항 이외에는 자유롭게 작성해도 무관합니다.

Model

- 정의할 모델 클래스의 이름은 Movie이며, 다음과 같은 정보를 저장합니다.

필드명	데이터 유형	역할
title	varchar(20)	영화 제목
audience	integer	관객 수
release_date	date	개봉일
genre	varchar(30)	장르
score	float	평점
poster_url	text	포스터 경로
description	text	줄거리

URL

- movies 앱은 다음 URL 요청에 맞는 역할을 가집니다.

URL 패턴	역할
/movies/	전체 영화 목록 페이지 조회
/movies/new/	새로운 영화 생성 페이지 조회
/movies/create/	단일 영화 데이터 저장
/movies/<pk>/	단일 영화 상세 페이지 조회
/movies/<pk>/edit/	기존 영화 수정 페이지 조회
/movies/<pk>/update/	단일 영화 데이터 수정
/movies/<pk>/delete/	단일 영화 데이터 삭제

View

- movies 앱은 다음 역할을 가지는 view 함수를 가집니다.

View 함수명	역할
index	전체 영화 데이터 조회 및 index.html 렌더링
new	장르 데이터 제공 및 new.html 렌더링
create	새로운 영화 데이터 저장 및 detail 페이지로 리다이렉트
detail	단일 영화 데이터 조회 및 detail.html 렌더링
edit	수정 대상 영화 데이터 조회 및 edit.html 렌더링
update	영화 데이터 수정 및 detail 페이지로 리다이렉트
delete	단일 영화 데이터 삭제 및 index 페이지로 리다이렉트

| Admin

- 모델 Movie를 Admin site에 등록합니다.
- Admin site에서 데이터의 생성, 조회, 수정, 삭제가 가능해야 합니다.

Template

- 사용 템플릿 목록
 1. base.html
 2. index.html
 3. detail.html
 4. new.html
 5. edit.html

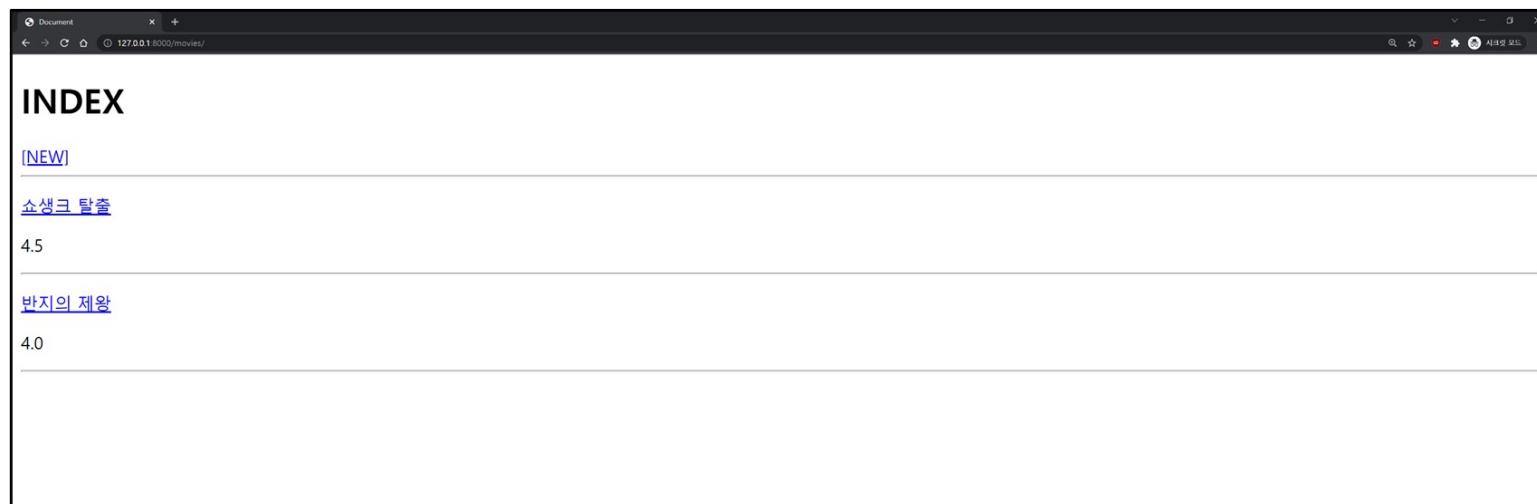
주의) 페이지 스크린 샷은 가독성을 위해 일부 확대 되었습니다.

| A. base.html

- 공통 부모 템플릿
 - 모든 템플릿 파일은 base.html을 상속받아 사용합니다.

B. index.html

- “전체 영화 목록 조회 페이지”
 - 데이터베이스에 존재하는 모든 영화의 목록을 표시합니다.
 - 적절한 HTML 요소를 사용하여 영화 제목 및 평점을 표시하며, 제목을 클릭 시 해당 영화의 상세 조회 페이지(detail.html)로 이동합니다.

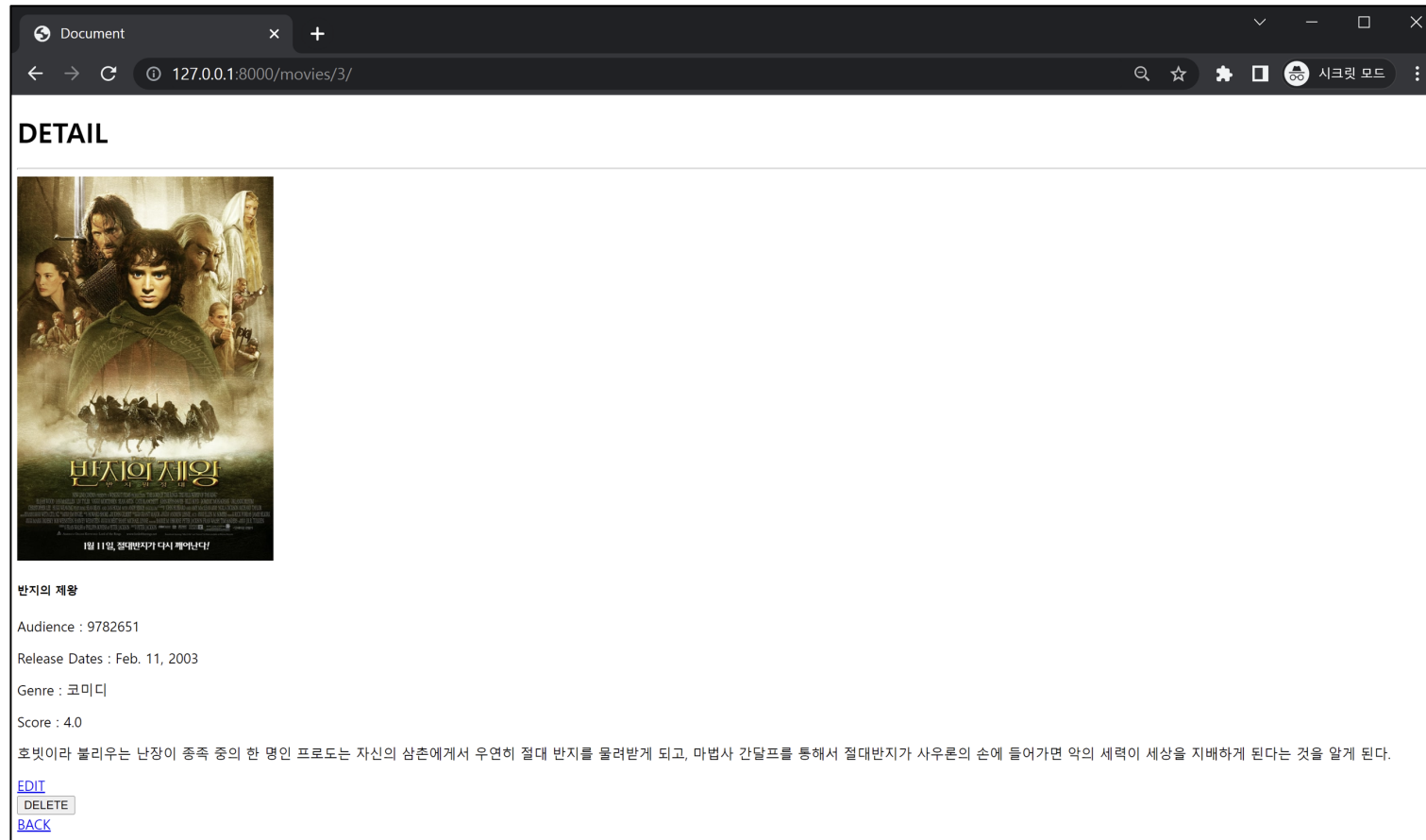


C. detail.html (1/2)

- “영화 상세 정보 페이지”
 - 특정 영화의 상세 정보를 표시합니다.
 - 해당 영화의 수정 및 삭제 버튼을 표시합니다.
 - 전체 영화 목록 조회 페이지(index.html)로 이동하는 링크를 표시합니다.

C. detail.html (2/2)

- “영화 상세 정보 페이지”

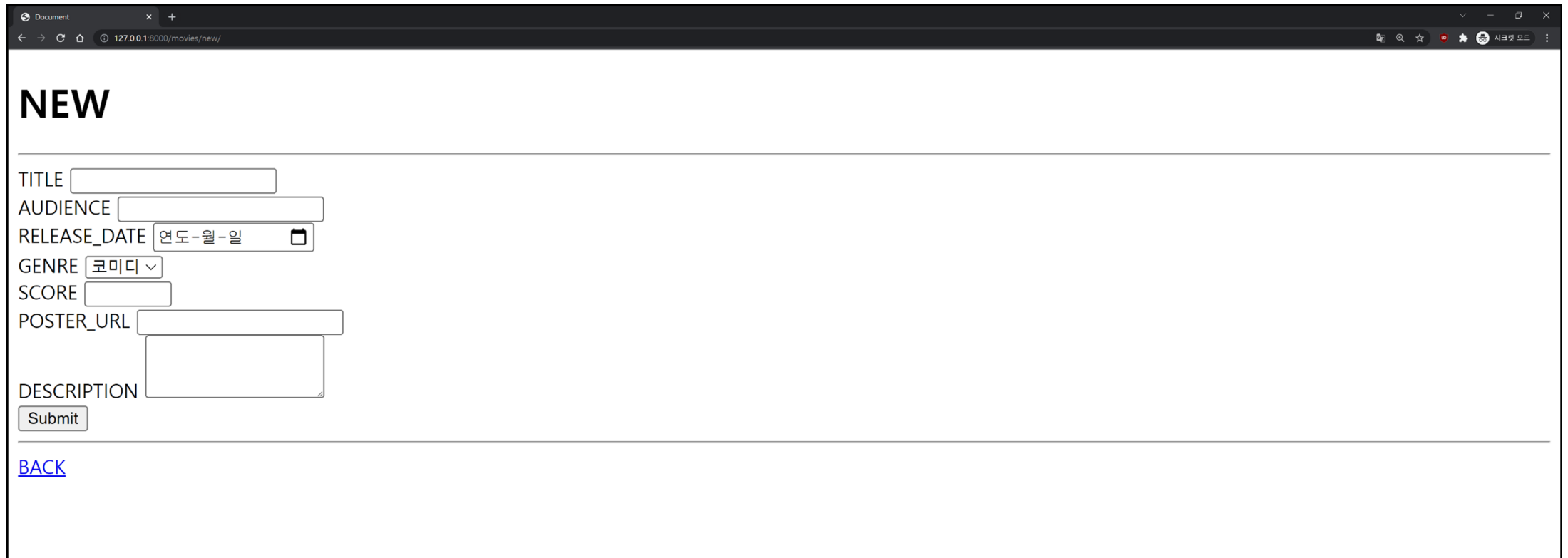


| D. new.html (1/2)

- “영화 생성 페이지”
 - 특정 영화를 생성하기 위한 HTML form 요소를 표시합니다.
 - 작성한 정보는 제출(submit)시 단일 영화 데이터를 저장하는 URL로 요청과 함께 전송됩니다.
 - 전체 영화 목록 조회 페이지(index.html)로 이동하는 링크를 표시합니다.

D. new.html (2/2)

- “영화 생성 페이지”



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:8000/movies/new/". The page content is titled "NEW" in large, bold, black letters. Below the title, there is a form with the following fields and controls:

- TITLE: A text input field.
- AUDIENCE: A text input field.
- RELEASE_DATE: A date input field with a placeholder "연도-월-일" and a calendar icon.
- GENRE: A dropdown menu with "코미디" selected.
- SCORE: A text input field.
- POSTER_URL: A text input field.
- DESCRIPTION: A text area.
- Submit: A button.

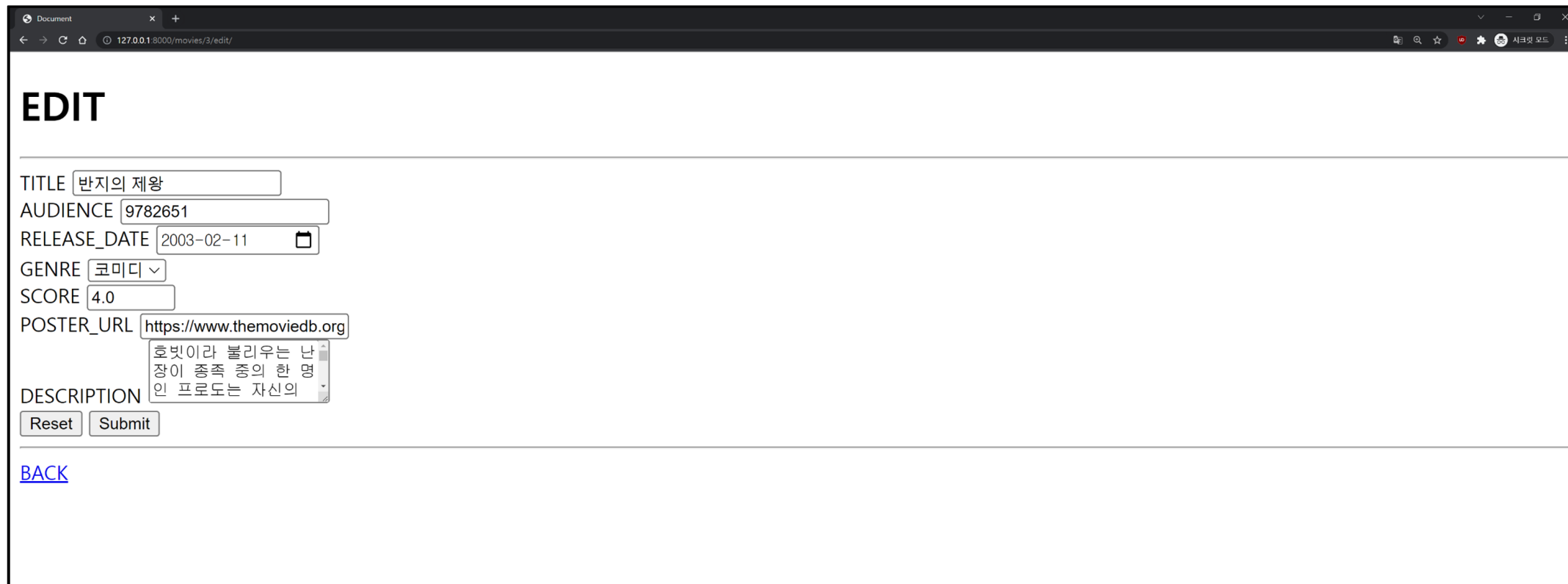
Below the form, there is a blue link labeled "BACK".

| E. edit.html (1/2)

- “영화 수정 페이지”
 - 특정 영화를 수정하기 위한 HTML form 요소를 표시합니다.
 - HTML input 요소에는 기존 데이터를 출력합니다.
 - Reset 버튼은 사용자의 모든 입력을 초기 값으로 재설정 합니다.
 - 작성한 정보는 제출(submit)시 단일 영화 데이터를 수정하는 URL로 요청과 함께 전송됩니다.
 - 영화 상세 정보 페이지(detail.html)로 이동하는 링크를 표시합니다.

E. edit.html (2/2)

- “영화 수정 페이지”



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:8000/movies/3/edit/". The page title is "EDIT". The form contains the following fields and values:

- TITLE: 반지의 제왕
- AUDIENCE: 9782651
- RELEASE_DATE: 2003-02-11
- GENRE: 코미디
- SCORE: 4.0
- POSTER_URL: https://www.themoviedb.org
- DESCRIPTION: 호빗이라 불리우는 난장이 종족 중의 한 명인 프로도는 자신의

At the bottom of the form, there are two buttons: "Reset" and "Submit". Below the form, there is a blue link labeled "BACK".

제출

제출 시 주의사항

- 제출기한은 금일 18시까지 입니다. 제출기한을 지켜 주시기 바랍니다.
- 반드시 README.md 파일에 단계별로 구현 과정 중 학습한 내용, 어려웠던 부분, 새로 배운 것들 및 느낀 점을 등을 상세히 기록하여 제출합니다.
 - 단순히 완성된 코드만을 나열하지 않습니다.
- 위에 명시된 요구사항은 최소 조건이며, 추가 개발을 자유롭게 진행할 수 있습니다.
- <https://lab.ssafy.com/>에 프로젝트를 생성하고 제출합니다.
 - 프로젝트 이름은 '프로젝트 번호 + pjт'로 지정합니다. (ex. **01_pjt**)
- 반드시 각 반 담당 교수님을 Maintainer로 설정해야 합니다.