

04 PJT

DB를 활용한 웹 페이지 구현

INDEX

- DB를 활용한 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
- Django ModelForm을 활용한 사용자 요청 데이터 유효성 검증
- Django Static files 관리 및 image file 업로드

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.
- .gitignore 파일을 추가하여 불필요한 파일 및 폴더는 제출하지 않도록 합니다.
- 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	varchar(50)	포스터 경로
description	text	줄거리
actor_image	image	대표 배우 이미지

URL

- movies 앱은 다음 URL 요청에 맞는 역할을 가집니다.

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

View

- movies 앱은 다음 역할을 가지는 view 함수를 가집니다.

View 함수명	역할	허용 HTTP Method
index	전체 영화 데이터 조회 및 index.html 렌더링	GET
create	create.html 렌더링 유효성 검증 및 영화 데이터 저장 후 detail.html 리다이렉트	GET & POST
detail	단일 영화 데이터 조회 및 detail.html 렌더링	GET
update	수정 대상 영화 데이터 조회 및 update.html 렌더링 유효성 검증 및 영화 데이터 수정 후 detail.html 리다이렉트	GET & POST
delete	단일 영화 데이터 삭제 및 index.html 리다이렉트	POST

| Admin

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

| Form

- A. Movie 모델의 데이터 검증, 저장, 에러메시지, HTML을 관리하기 위해 ModelForm을 사용합니다.
- B. genre 필드
 - I. select element를 출력해 코미디, 공포, 로맨스 장르 데이터를 선택할 수 있도록 합니다.
- C. score 필드
 - I. input element의 type은 number로 설정합니다.
 - II. input element attribute 중 step은 0.5, min은 0, max는 5로 설정합니다.
- D. release_date 필드
 - I. input element의 type은 date로 설정합니다.

| Template

- 사용 템플릿 목록
 - A. base.html
 - B. index.html
 - C. detail.html
 - D. create.html
 - E. update.html

A. base.html

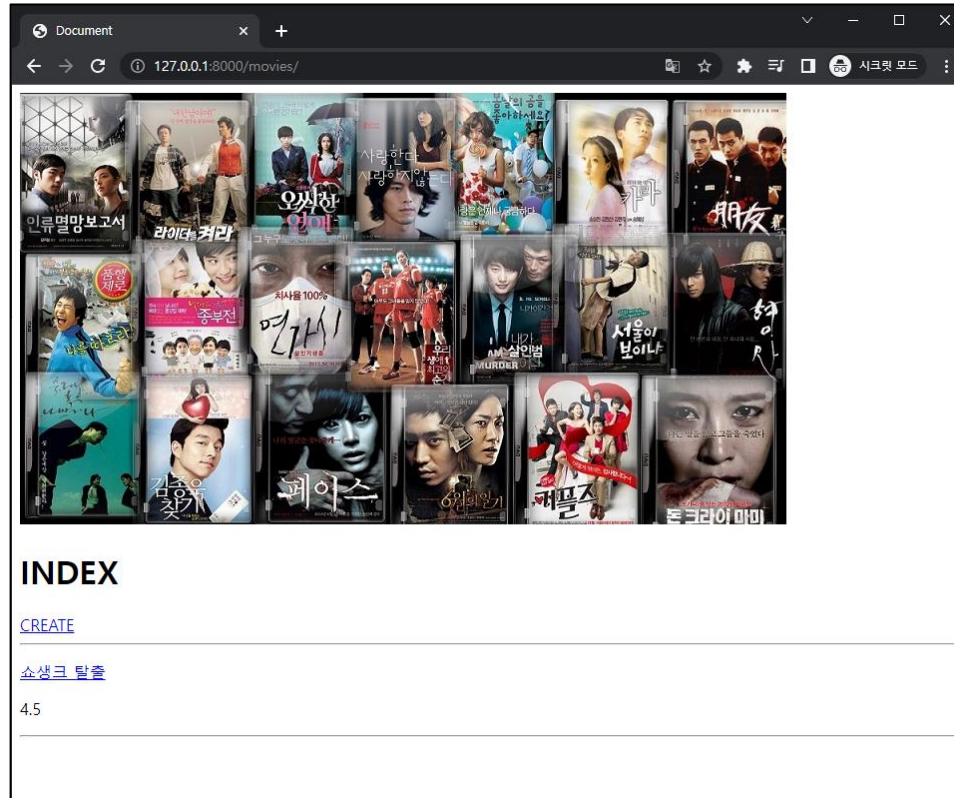
- 공통 부모 템플릿
 - 모든 템플릿 파일은 base.html을 상속받아 사용합니다.
 - 주어진 header.jpg를 화면 상단에 보여줍니다.

| B. index.html (1/2)

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

B. index.html (2/2)

- “전체 영화 목록 조회 페이지”

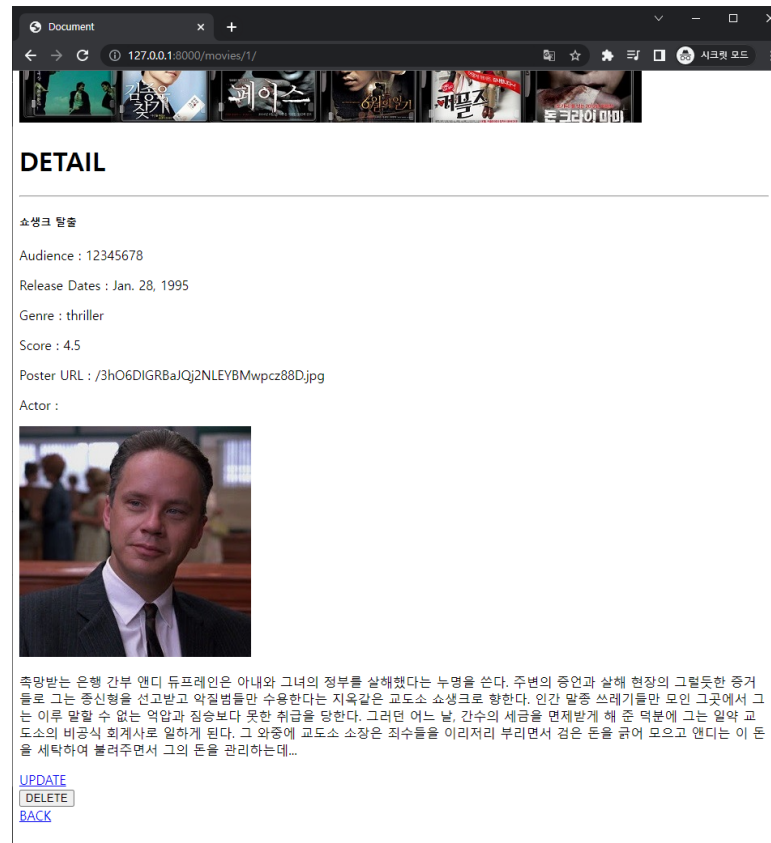


C. detail.html (1/2)

- “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”

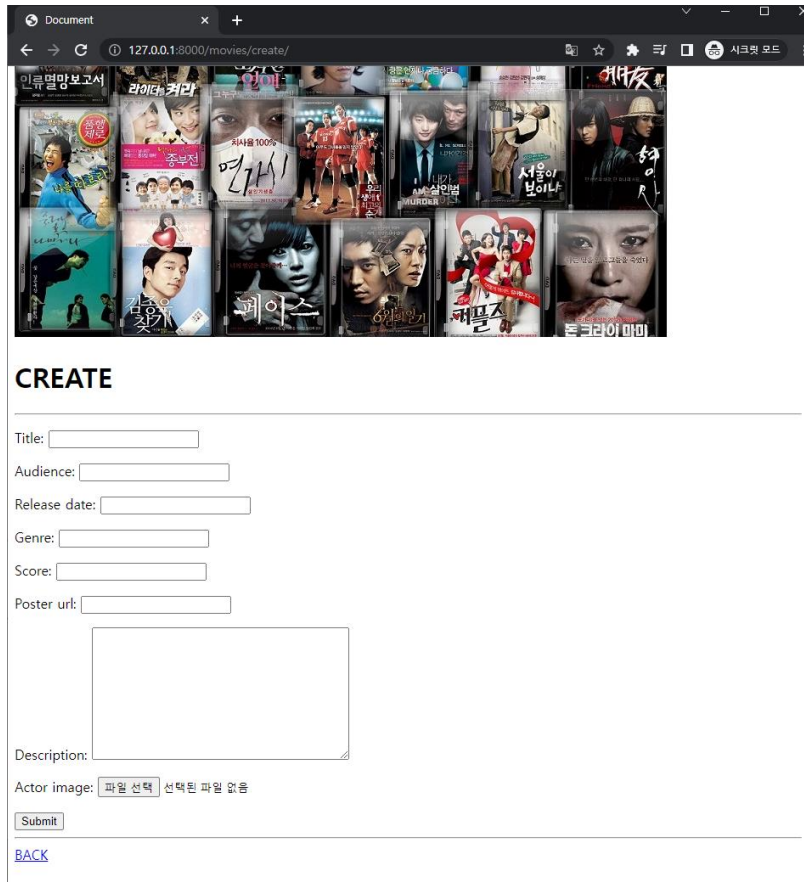


| D. create.html (1/2)

- “영화 생성 페이지”
 - 특정 영화를 생성하기 위한 HTML form 요소를 표시합니다.
 - 표시되는 form은 Movie 모델 클래스에 기반한 ModelForm이어야 합니다.
 - 작성한 정보는 제출(submit)시 단일 영화 데이터를 저장하는 URL로 요청과 함께 전송됩니다.
 - 전체 영화 목록 조회 페이지(index.html)로 이동하는 링크(back)를 표시합니다.
 - actor_image에 해당하는 이미지는 직접 업로드할 수 있습니다.

D. create.html (2/2)

- “영화 생성 페이지”



Document x +

127.0.0.1:8000/movies/create/

CREATE

Title:

Audience:

Release date:

Genre:

Score:

Poster url:

Description:

Actor image: 선택된 파일 없음

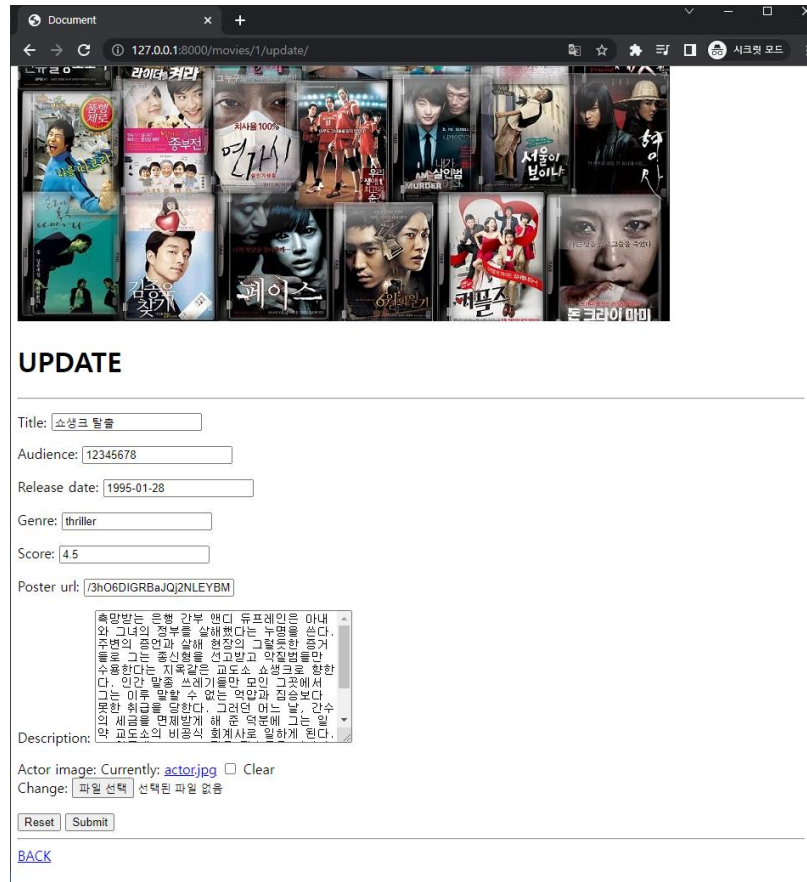
[BACK](#)

| E. update.html (1/2)

- “영화 수정 페이지”
 - 특정 영화를 수정하기 위한 HTML form 요소를 표시합니다.
 - 표시되는 form은 Movie 모델 클래스에 기반한 ModelForm이어야 합니다.
 - 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 링크(back)를 표시합니다.

E. update.html (2/2)

- “영화 수정 페이지”



Document x +

127.0.0.1:8000/movies/1/update/

UPDATE

Title:

Audience:

Release date:

Genre:

Score:

Poster url:

Description:

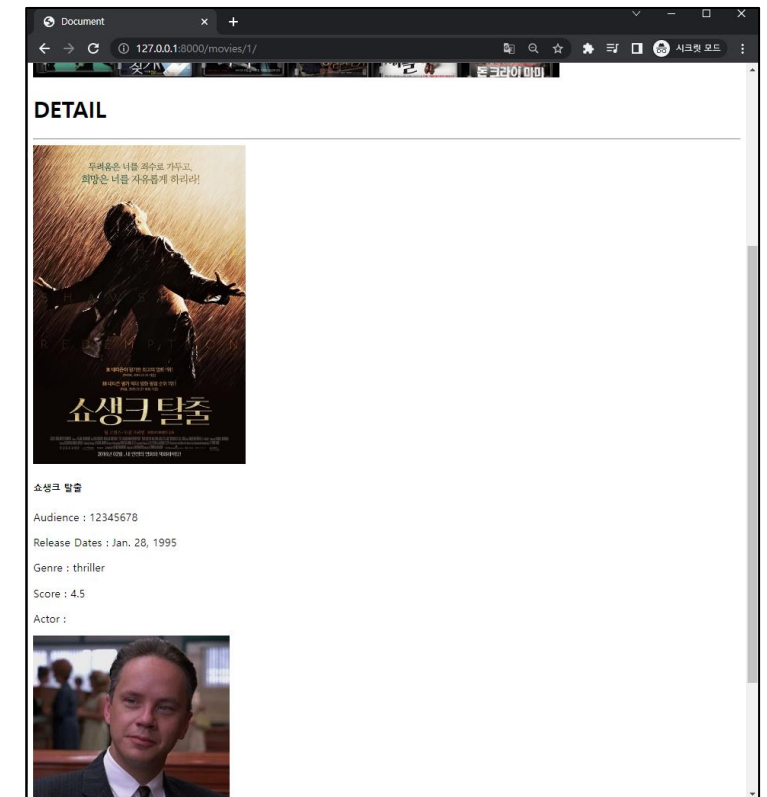
Actor image: Currently: ☐ actor.jpg ☐ Clear

Change: ☐ 파일 선택 ☐ 선택된 파일 없음

[BACK](#)

F. 선택과제

- [TMDB API](#)를 참고하여 poster_url 필드를 활용하여 detail 페이지에서 포스터 이미지를 보여줍니다.
- poster_url 데이터는 pjt01의 데이터 또는 TMDB의 데이터를 활용합니다.

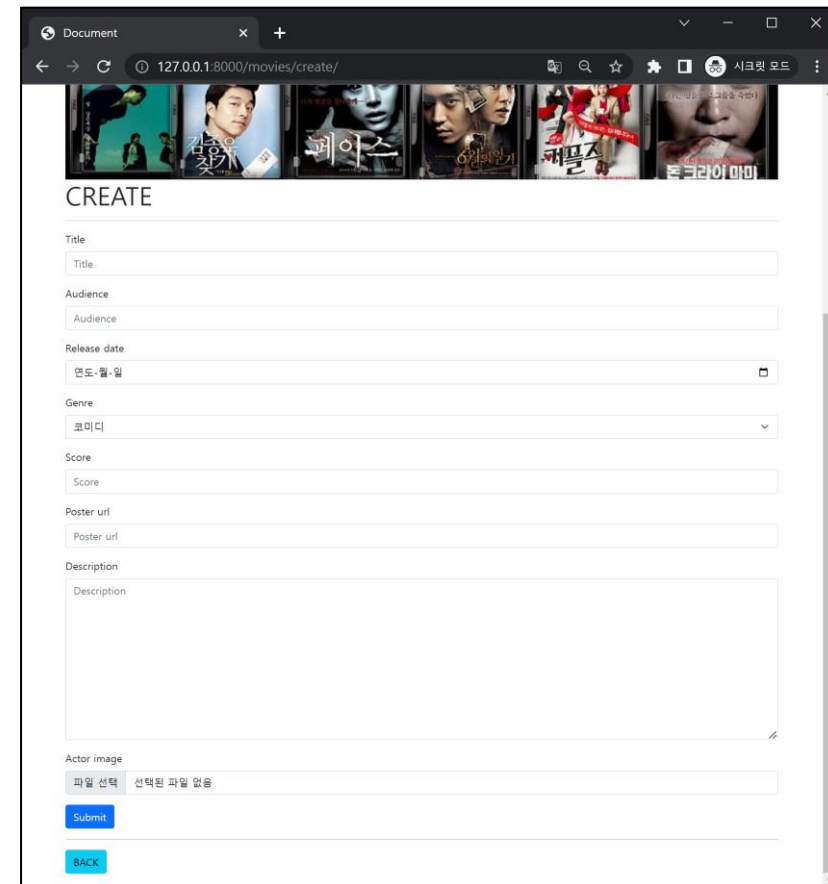
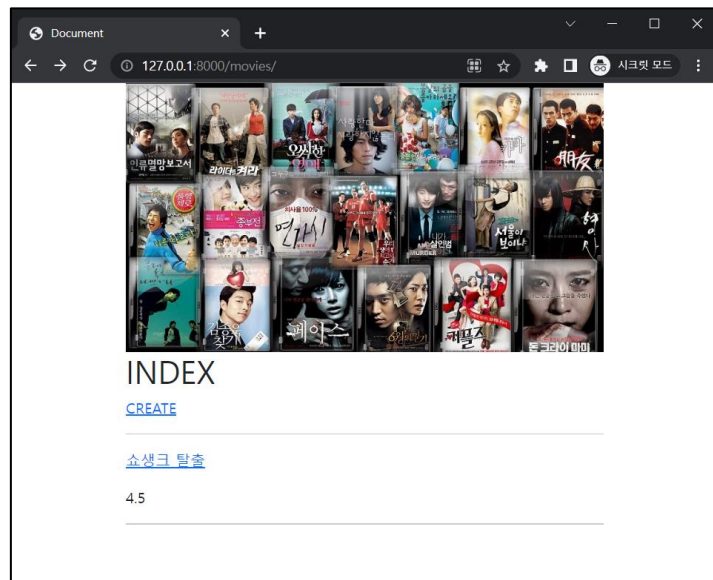


F. 선택과제

- Bootstrap 또는 기타 라이브러리를 활용하여 페이지를 꾸밉니다.
 - Bootstrap 사용을 위한 코드(cdn)는 어디에 위치하면 좋을지 생각해봅니다.
 - Detail페이지에는 Bootstrap Card Component를 사용해봅니다.

F. 선택과제

- 예시



F. 선택과제

- 예시

Document

127.0.0.1:8000/movies/1/update/

UPDATE

Title

소생크 탈출

Audience

12345678

Release date

1995-01-28

Genre

공포

Score

4.5

Poster url

/3h0d0dIGR8aIQ2NLEY8Mvpcz88D.jpg

Description

죽어가는 은행 강부 연디 듀르레인은 아내와 그녀의 정부를 살해했다는 누명을 쓴다. 주변의 증언과 실제 현장의 그칠듯한 증거들로 그는 중신형을 선고 받고 억울함을만 수형한다는 지옥같은 교도소 소생크로 향한다. 인간 밑둥 쓰러기들만 모인 그곳에서 그는 이후 말할 수 없는 억압과 정승보다 못한 취급을 당한다. 그러던 어느 날, 간수의 세금을 강제받게 된 순간 역으로 그는 밀어 고르스의 비공식 회계사로 일하게 된다. 그 와중에 교도소 소장은 죄수들을 이리저리 부리면서 깊은 돈을 끌어 오르고 연디는 이 돈을 세탁하여 돌려주면서 그의 돈을 관리하는데...

Actor image

Currently: actor.jpg ☐ Clear

Change:

파일 선택 선택된 파일 없음

Submit Cancel

BACK

Document

127.0.0.1:8000/movies/1/

DETAIL

소생크 탈출

Audience : 12345678

Release Dates : Jan. 28, 1995

Genre : thriller

Score : 4.5

Actor :

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.