

06) Django Intro

1. 목표

- 데이터를 생성, 조회, 삭제, 수정할 수 있는 Web Application 제작
- Python Web Framework를 통한 데이터 조작
- Object Relational Mapping에 대한 이해
- Template Variable을 활용한 Template 제작
- 영화 추천 사이트의 영화 정보 데이터 관리

2. 준비 사항

1. (필수) Python Web Framework

- Django 2.1.x
- Python 3.6.x

2. (선택) 샘플 영화 정보

- [링크](#)
 - 대표적인 영화 총 6개의 정보가 포함된 `data.csv`가 있습니다.
 - 1주차에 진행 하였던 프로젝트 코드들을 통해 여러분이 원하는 데이터를 만들어서 실습할 수 있습니다.

3. 요구 사항

1. 데이터베이스

- ORM을 통해서 작성될 클래스의 이름은 `Movie`이며, 다음과 같은 정보를 저장합니다.

| 필드명 | 자료형 | 설명 |
|-------------|---------|-------------|
| title | String | 영화명 |
| audience | Integer | 누적 관객수 |
| genre | String | 장르 |
| score | Float | 평점 |
| poster_url | Text | 포스터 이미지 URL |
| description | Text | 영화 소개 |

2. 페이지

1. 영화 목록

1. **(필수)** 해당 페이지에 접근하는 URL은 `/movies/` 입니다.
2. **(필수)** 데이터베이스에 존재하는 모든 영화의 목록이 표시 되며, 각 영화의 `title`, `score` 가 표시 됩니다.
3. **(필수)** `title` 을 클릭 시, 해당 `영화 정보 조회` 페이지로 이동합니다.

2. 영화 정보 조회

1. **(필수)** 해당 페이지에 접근하는 URL은 `/movies/1/`, `/movies/2/` 등 이며, 동적으로 할당되는 부분이 존재합니다. 동적으로 할당되는 부분에는 데이터베이스에 저장된 영화 정보의 Primary Key 가 들어갑니다.
2. **(필수)** 해당 Primary Key를 가진 영화의 **모든 정보**가 표시됩니다.
3. **(필수)** 영화 정보의 최하단에는 `목록`, `수정`, `삭제` 링크가 있으며, 클릭 시 각각 `영화 목록`, 해당 `영화 정보 수정 Form`, 해당 `영화 정보 삭제` 페이지로 이동합니다.

3. 영화 정보 삭제

1. **(필수)** 해당 페이지에 접근하는 URL은 `/movies/1/delete/`, `/movies/2/delete/` 등 이며, 동적으로 할당되는 부분이 존재합니다. 동적으로 할당되는 부분에는 데이터베이스에 저장된 영화 정보의 Primary Key가 들어갑니다.
2. **(필수)** 해당 Primary Key를 가진 영화 정보를 데이터베이스에서 삭제합니다.
3. **(필수 - 택1)**
 - '영화 정보 삭제가 완료되었습니다.' 메시지가 표시되며, `영화 목록` 으로 이동하는 링크가 표시 됩니다.
 - `영화 정보 목록` 페이지로 Redirect 합니다.

4. 결과 예시

Python Web Framework를 활용해 작성한 모든 파일을 `06_django` 디렉토리에 위치하도록 합니다.

결과물은 반드시 `README.md` 으로 활용 하였던 내용을 정리 해주세요.

```
06_django/  
...  
...  
...  
README.md
```