

도서관대출_엘리스

2팀 이금홍

프로젝트 수행 과정

UI 작업 & DB 구성

- 예시 Template을 보고 비슷한 Bootstrap Template을 다운 받아서 수정
- 예시 Template을 보고 DB 구성

선택 기능 구현

- session, RegExp, ORM

필수 기능 구현

- 로그인, 로그아웃, 회원가입
- 메인 페이지, 대여하기, 반납하기, 책소개

배포

- Azure 설정 및 배포
- <http://kdt-1st-project-38.koreacentral.cloudapp.azure.com/>

[38.koreacentral.cloudapp.azure.com/](http://kdt-1st-project-38.koreacentral.cloudapp.azure.com/)

배운 점

01

Jinja2 사용법
- {{ 변수명 }}, {% %}

02

Flask 사용법
- <https://wikidocs.net/book/4542>

03

ORM & CRUD 사용법
- Flask-SQLAlchemy & MySQL

1. 회원 가입 – 이름은 한글, 영문으로만 입력
– 비밀번호는 영문, 숫자, 특수문자 중 2종류 이상을 조합하여 최소 10자리 이상
또는 3종류 이상을 조합하여 최소 8자리 이상의 길이
2. 로그인 & 로그아웃 – session으로 관리
3. 메인 페이지 – DB 상에 존재하는 모든 책 정보
– 책 이름 클릭 시 책 소개 페이지로 이동
– 한 페이지 당 8권 책 표기 (페이지네이션)
4. 대여하기 – 대여하기 버튼 클릭 시 해당 책 대여 완료
– 재고가 없을 경우 대여 불가 메시지 출력, 있을 경우 책 권수 - 1 표기
– 이미 빌린 책이라면 메시지 출력 및 대여 불가
5. 반납하기 – 로그인 한 유저가 대여한 책들 표시
6. 책 소개 – 해당 책에 대한 정보들을 상세히 출력
– 댓글 작성 가능 (유정 당 1개씩) 별점 부여(1 ~ 5)
7. 대여기록 – 유저가 반납한 책들 표시

어려웠던 점

1. Flask-SQLAlchemy 와 SQLAlchemy는 다르다

- 구글링 시 다르다는 것을 인지하지 못해서 코드 작성 시 많은 시간을 허비함

2. 실행되는 .py파일에서는 상대경로를 사용 할 수 없다.

- `from . import models` 불가 // `from models import *` 가능

3. redirect, url_for 함수의 사용법

- 변수를 넘기는 법, 파라미터를 넘기는 법 검색하고 적용하는 시간이 생각보다 길었음

4. 코치님

- 있으셔서, 프로젝트 가능, 안 계셨으면 막히는 부분에서 시간을 다 허비하던지...

혹은 틀린 방향으로 프로젝트를 진행 했을 확률이 높음

Q&A

감사합니다.

