

과제 5

2025.06.05



LAB5

- 한 학기동안 개인이 좋아하는 영화를 검색할 수 있는 웹사이트 구축을 목표로 함
- 본 과제에서는 다음과 같은 파일들 제출(1개라도 없으면 0점)

(단, 아래 파일들은 필수 파일이고 동적 페이지 구성을 어떻게 하는가에 따라 추가적인 파일이 생길 수 있음)

- ① index.js (조교가 node index.js로 실행할 파일)
 - 서버 실행이 안되면 LAB5 과제 0점
- ② product.db 파일 (기존의 product.json 파일 대체)
 - ❖ 주의: 영화 포스터 이미지의 경우 불러오는 방법에 따라 추가적으로 이미지 파일들을 과제 제출 시 같이 포함시켜야 함. (구현 방식에 따라 다름)
 - 이미지가 올바르게 로드되지 않는 경우 감점 (-10 점)
- ③ comment.json (영화별 후기 정보 포함)

LAB5

- Requirements (200점 만점) - 기본 구성
 - localhost:3000에 보이는 컴포넌트들은 width가 1020px 내에 존재해야 하고 가운데에 존재해야 함
(10pt, 1개의 컴포넌트라도 1020px을 넘어가거나 가운데에 존재하지 않을 시 0점)
 - 제출 페이지 간의 linking이 있어야 함 (15pt, 한 개라도 빠질 시 0점)
 - ✓ 메인페이지(localhost:3000) ↔ 로그인 페이지(localhost:3000/login)
 - ✓ 메인페이지(localhost:3000) ↔ 회원가입 페이지(localhost:3000/signup)
 - ✓ 로그인 페이지(localhost:3000/login) ↔ 회원가입 페이지(localhost:3000/signup)
 - 메인 페이지, 로그인 페이지, 회원가입 페이지, 영화 설명 페이지(뒤에 설명)에 대해 1개의 페이지라도 css가 입혀지지 않았다면 LAB5 과제 0점 처리

LAB5

- Requirements (200점 만점) - Database 활용
 - LAB4의 요구사항과 구현 방식은 비슷하지만 product.json과 Fetch API 사용불가
 - **product.db**: 사이트에 보여줄 영화 정보를 DB로 저장 및 관리 (30pt)
 - 영화 정보를 보여주기 위해서 테이블은 다음과 같은 field값 필수포함 해야함:
 - ✓ movie_id(PK, AI 적용), movie_image(영화 이미지 경로), movie_title, movie_overview, movie_release_date, movie_rate
(각 component 마다 5점)
 - ✓ 제공해드리는 movies_metadata.csv 파일의 영화 정보 이용
 - 전체 영화 개수는 최소 20개로 지정 (20개 미만시 0점 처리)
 - ✓ product.db 구성 방법은 SQLite 실습 영상 참고
 - 키워드 검색을 통해 영화를 검색하면 해당 영화를 필터링 해서 보여줌
 - ✓ localhost:3000에 db파일에 저장된 정보를 불러와야 함 (20pt)
 - ✓ 영화 필터링
 - 키워드 검색 기능 (25pt)
 - 카테고리 필터링 (25pt)

LAB5

- Requirements (200점 만점) - Database 활용 (영화 설명 페이지)
 - 영화를 클릭하면 영화에 대한 상세 페이지로 redirection 해 줌
(localhost:3000 → localhost:3000/movies/:movie_id)

아래와 같은 영화의 상세 정보를 localhost:3000/movies/:movie_id에서 표시해줌
(25점, 한 개라도 빠지면 0점)

- movie_id
- movie_image
- movie_title
- movie_overview
- movie_release_date
- movie_rate

LAB5

- Requirements - File I/O 활용
 - localhost:3000/movies/:movie_id은 추가적으로 영화에 대한 후기도 표시해야 함 (File I/O 활용)
 - 사용자가 새로운 후기를 남기면 새로 고침 했을 때 후기가 추가되는 기능 (25점)
 - 새로운 후기를 남기면 파일 내용이 comment.json에도 업데이트 되는 기능 (25점)
 - comment.json은 requirements를 해결할 수 있게 개인적으로 구성

LAB5 - 예시

width 1020px 이내

가운데 정렬

인프밍 영화 정보 사이트입니다.

[메인페이지](#) [로그인](#) [회원가입](#)

모든 페이지 간의
linking 필요

키워드를 입력하세요.

Filter results

Movies

정렬 기준

- ☐ 평점 내림차순
- ☐ 평점 오름차순
- ☐ 개봉 내림차순
- ☐ 개봉 내림차순



Toy Story

1995-10-30



Jumanji

1995-12-15

LAB5 - 예시

인프밍 영화 정보 사이트입니다.

[메인페이지](#) 로그인 회원가입

DB로 저장 및 관리하여
정보 display

키워드를 입력하세요.

Filter results

Movies

정렬 기준

- ☐ 평점 내림차순
- ☐ 평점 오름차순
- ☐ 개봉 내림차순
- ☐ 개봉 내림차순



Toy Story
1995-10-30



Jumanji
1995-12-15

영화 필터링
(DB에 추가한 field
요소 활용)

LAB5 - 예시 (영화 상세페이지)


인프밍 영화 정보 사이트입니다.


[메인페이지](#) [로그인](#) [회원가입](#)


DB에 저장되어 있는
영화 정보 display




영화 id: 1

 **영화 제목** Toy Story

 개봉일: 1995-10-30

 평점: 7.7

 줄거리: Led by Woody, Andy's toys live happily in his room until Andy's birthday brings Buzz Lightyear onto the scene. Afraid of losing his place in Andy's heart, Woody plots against Buzz. But when circumstances separate Buzz and Woody from their owner, the duo eventually learns to put aside their differences.

→ 미리 만들어 놓은 파일에 존재하는 후기 불러오기

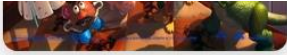
영화 후기

재밌었어요!

명작!

굳

LAB5 - 예시 (영화 상세페이지)



— 영화 후기

재밌었어요!

명작!

군

최고의 영화입니당 :)

재미 없어요.

등록하기!



새로운 후기 추가 시 업데이트 됨을 보여야 함

— 영화 후기

재밌었어요!

명작!

군

최고의 영화입니당 :)

재미 없어요.

후기를 작성하세요

등록하기!

Deliverables

- 제출 날짜와 시간(06/19(목) 아침 9시)을 지난 제출물은 받지 않음
 - LearnUs에는 zip 파일 제출
 - 채점과는 관련 없지만 개인적으로 github에 올리는 것을 추천
- 채점 기준 (200점)
 - Requirements에서 요구하는 사항 모두 만족 필요
- 과제 upload시 파일명은 아래와 같이 표시
 - 파일명이 다를 시에는 점수 공제

	File name	Due
LearnUs (제출)	LAB5_<student_id>.zip	06/19 (목) 아침 9시

Q&A

- 06/10(화) 수업시간에 과제 관련 질의응답 시간을 가질 예정입니다.
- 과제에 대한 질문이 있으면 수업 조교에게 문의 바랍니다.
 - 엄계현(KaeHyun Um) : khyun33@yonsei.ac.kr
 - 오유솔(YuSol Oh) : wina0817@yonsei.ac.kr
 - 김민지(MinJi Kim) : minjikirn_ai@yonsei.ac.kr