lowdb

-> 사진보고 설치

lowdb: sql명령어 없이 데이터를 저장하는 기능을 가진 모듈, 데이터는 지정한 json에 저장된다.

[ lowdb 세팅 방법 ]

// 회원가입 관련 모듈 불러오기

const low = require('lowdb'); // lowdb를 이용해서 데이터 저장

const FileSync = require('lowdb/adapters/FileSync'); // 동기 방법으로 저장

const adapter = new FileSync('db.json'); // db.json이라는 파일에 데이터 저장

const db = low(adapter); // db키워드를 이용해서 데이터 저장

// 데이터가 db.json의 users,topics위치에 저장되도록 설정

db.defaults({users:[], topics:[]}).write();

module.exports=db;

[json파일 형태 예시]

{

"users": [

{

"Private": "3muLcjfbo",

"Id": "af",

"Password": "af",

"Nickname": "af"

}

],

"topics": [

{

"post\_id": "\_D-HrxFIt",

"title": "rwgr",

"description": "wrgge",

"user\_private": "O8Qf3XU1m",

"user\_nickname": "af"

}

]

}

[ lowdb 명령어 ]

// json에서 데이터가 담긴 객체를 생성

db.get('topics').push({

post\_id: 123,

title: hello,

description: nodejs,

user\_private: qwedddww,

user\_nickname: hman

}).write();

response.redirect(`/page/${post\_id}`);

// json에서 해당 객체를 찾아서 반환

db.get('topics').find({post\_id:id}).value();

// json에서 해당 객체를 찾아서 다음과 같이 수정

db.get('topics').find({post\_id:id}).assign({

title:filteredTitle,

description:description

}).write();

// json에서 해당 객체를 찾아서 삭제

db.get('topics').remove({post\_id:id}).write();