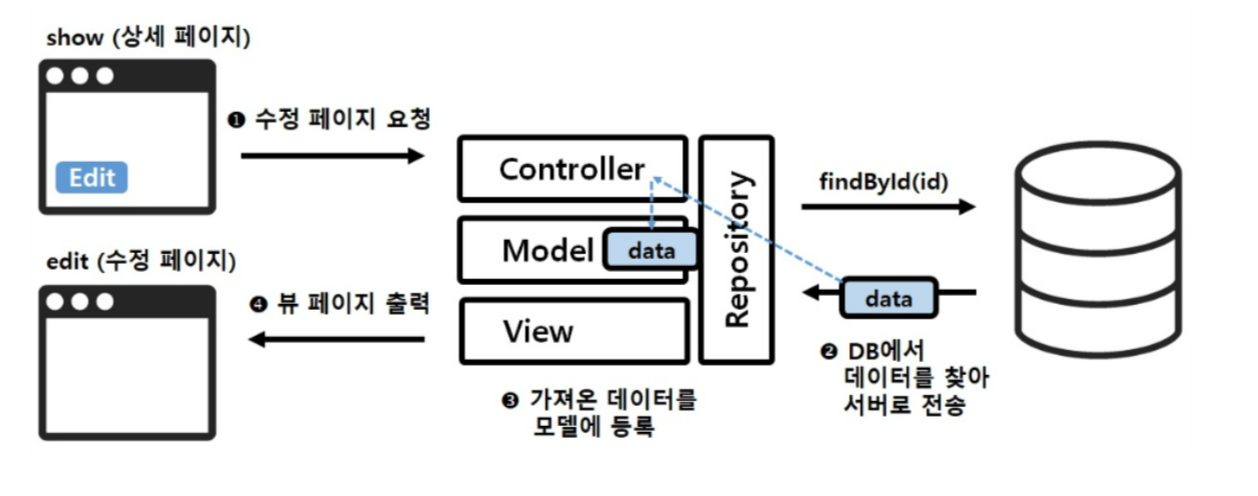
게시글 수정하기 update

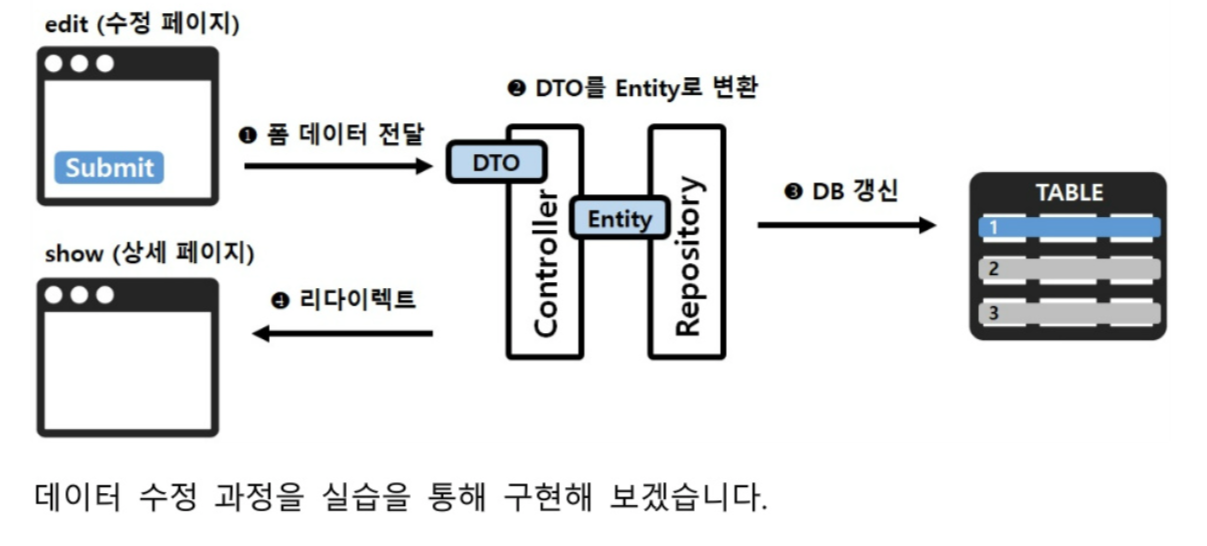
1. 수정 페이지를 만들고 기존의 데이터를 불러오기
2. 데이터를 수정해 DB에 반영한 후 결과를 볼 수 있게 <상세 페이지>로 리다이렉트하기

1단계 동작원리

* 상세페이지에서 edit 버튼 클릭
* 요청받은 컨트롤러는 해당 글의 id 로 DB에서 데이터를 찾아 가져옴
* 컨트롤러는 가져온 데이터를 뷰에서 사용할 수 잇도록 모델에 등록
* 모델에 등록된 데이터를 수정 페이지 에서 보여줌 -> 사용자가 내용을 수정할 수 있는 상태가 된다.

2단계 동작원리

* 폼 데이터를 DTO에 담아 컨트롤러에서 받는다
* DTO를 엔티티로 변환
* DB에서 기존 데이터를 수정 데이터로 갱신
* 상세페이지로 리다이렉트진행



수정페이지 제작

* 상세페이지에 edit링크를 걸기

Edit 요청을 받아 데이터를 가져오기

1. 수정 요청을 받아 처리할 edit() 메서드를 작성하고 반환할 수정 페이지를 articles 디렉토리 안에 edit.mustache 파일로 설정
2. url 요청을 방는 @GetMapping()을 작성 (“/articles/{id}/edit”) 컨트롤러에서 url 변수를 사사용할 때 중괄호를 무조건 한 개만

수정할 데이터 가져오기

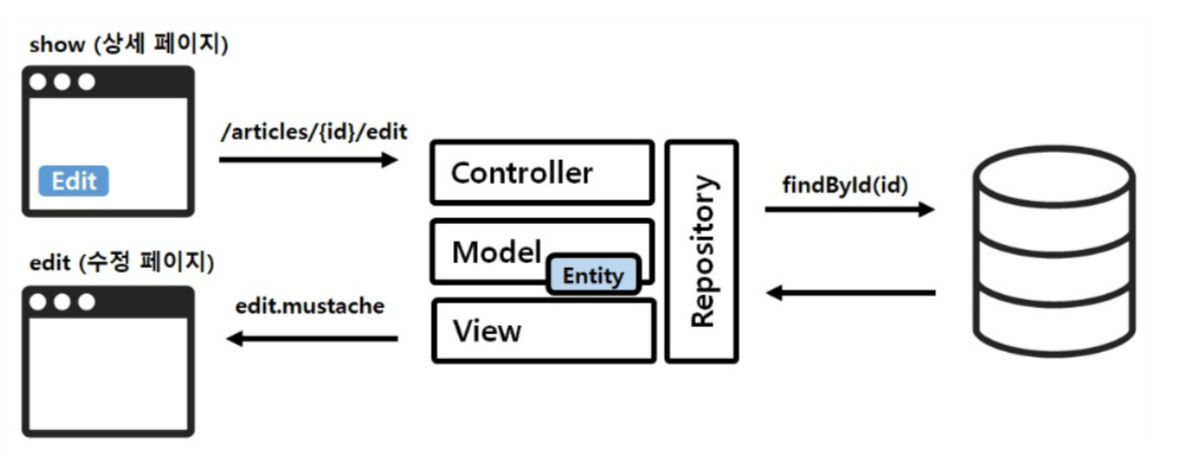
1. DB에서 데이터를 가져올 때는 repository를 이용. 따라서 articleReposirtory의 findById(id) 메소드로 데이터를 찾아온다 찾지 못하면 null 반환, 찾으면 Article 타입의 articleEntity로 저장

모델에 데이터를 등록하기

1. 모델을 사용하기 위해 메소드의 매개변수로 model 객체를 받는다
2. addAttribute() 메소드로 모델에 데이터를 등록한다. article이라는 이름으로 앞에서 가져온 articleEntity를 등록. 이렇게 한다면 DB에서 가져온 데이터를 article이라는 이름으로 뷰 페이지에서 사용이 가능!

수정폼을 제작하기 (mustache)

이번장 요약

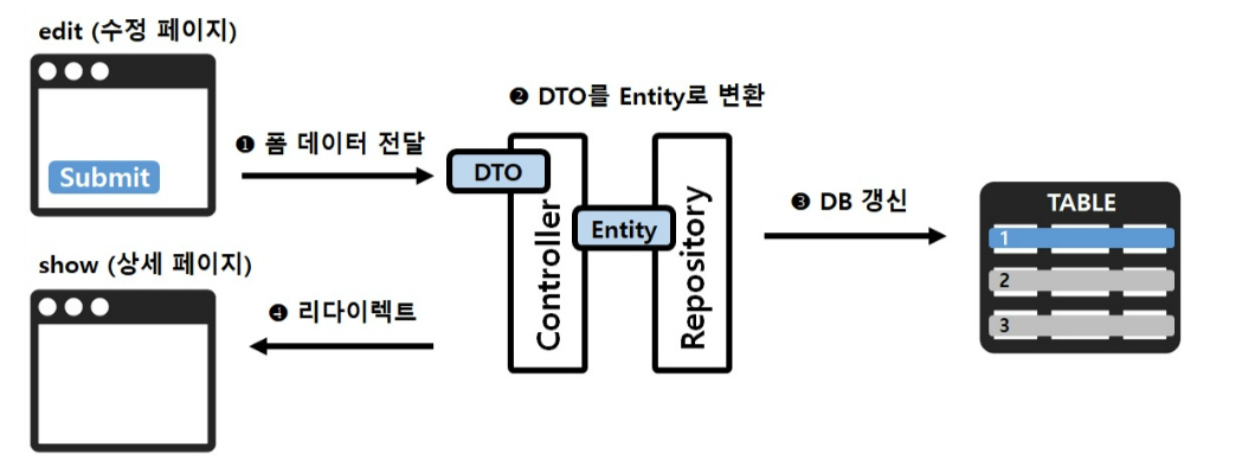


1분퀴즈

수정페이지 처리흐름 순서

1. 클라이언트로부터 데이터 수정 요청을 받는다
2. 수정 요청 데이터를 DB에서 찾는다
3. 수정 요청 데이터를 모델에 등록한다
4. 뷰 페이지에 수정할 데이터를 함께 보여준다

수정 데이터를 DB에 갱신하기

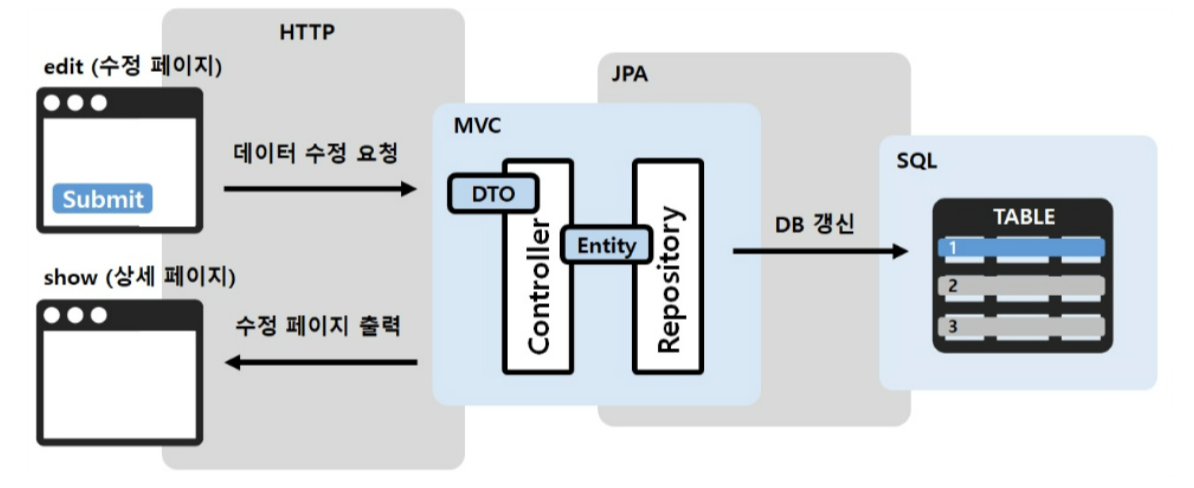


MVC : 서버 역할을 분담하여 처리하는 기법

JPA : 서버와 DB간의 소통에 관여하는 기술

SQL : Db 데이터를 관리하는 언어

HTTP : 데이터를 서버로 전송할 때 사용하는 통신규약



HTTP 의 대표적인 메소드

GET : 데이터 조회 요청

POST : 데이터 생성 요청

PATCH : 데이터 수정 요청

DELETE : 데이터 삭제 요청

데이터의 조회, 생성, 수정, 삭제는 데이터 관리에서 가장 기본이 되는 동작이다. 이를 간단히 해서 CRUD 라고도 한다.

데이터를 DTO에 담고 이를 DB에 저장하는 과정

1. DTO를 엔티티로 변환
2. 엔티티를 DB에 저장
3. 결과 페이지로 리다이렉트 진행