게시판 구현 - 글쓰기

controller

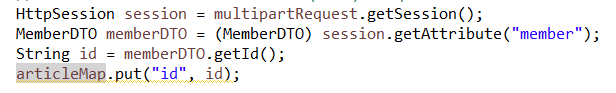


ResponseEntity를 리턴해 페이지 상태와 헤더를 담아 전송할 수 있음

파일 업로드를 위해 MultipartHttpServletRequest 객체 사용

.getParameterNames()는 Enumeration 리턴

.getParameterNames()로 받은 파라미터들을 Map<name,value> articleMap에 저장



session 영역에 저장된 로그인 MemberDTO.id를 articleMap에 저장

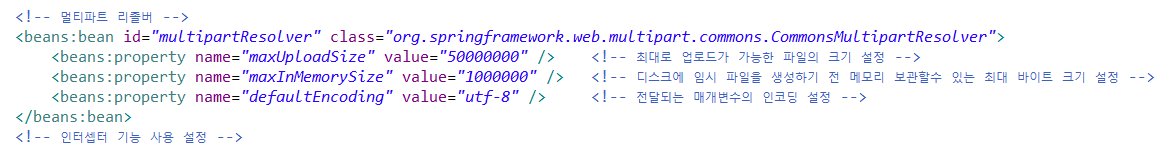
파일 업로드

pom.xml



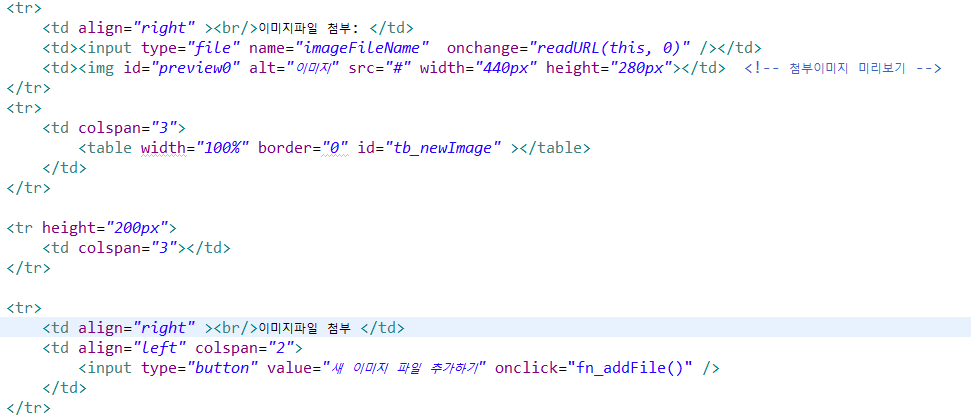
파일업로드를 위한 디펜던시

spring-context



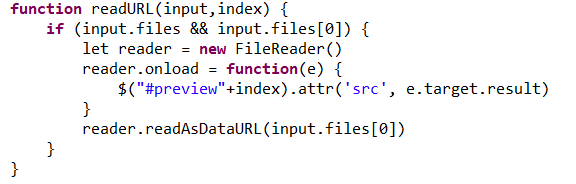
multipartResolver Bean 주입

jsp



<input type=”file”> 이용해 파일 입력

onchange : 값이 변경되었을 때 호출



if (input.files && input.files[0])

input.files 는 FileList를 반환함, input.files[0]는 첫 번째 파일이란 의미

즉, 파일이 존재하고 첫 번째 파일이라는 뜻

파일을 읽어오기 위해 FileReader 객체 선언

reader.onload = function(e) : 리더가 가동하면 그 값으로 function(e) 실행

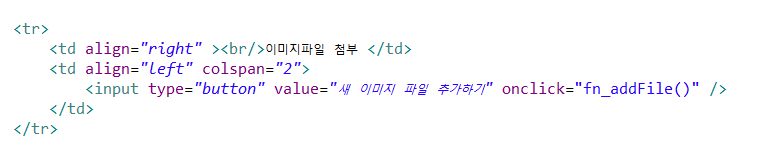
e.target : 이벤트가 실행된 객체를 가리킴

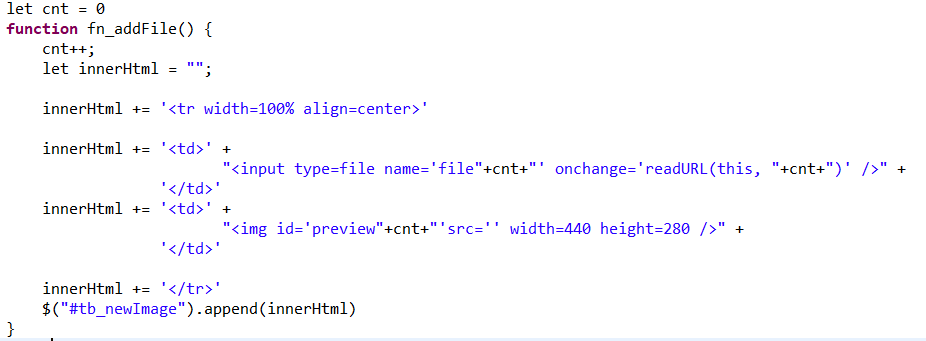
img 태그인 #preview+index의 src에 그 결과값 입력

e.target.result

즉, reader.readAsDataURL(input.files[0]) 실행 시 reader.onload function 발생





이미지 추가할 시 input마다 fileList[0]



Iterator fileNames에 전송된 파일이름 저장(name : imageFileName, file1,2…)

MultipartFile객체 mFile에 fileName에 저장된 파일 저장

originalFilename에 실제 저장된 파일 이름 저장

메서드 리턴값 filelist에 originalFileName 저장하고

File file = new File(ARTICLE\_IMAGE\_REPO +"\\"+ fileName);

파일 디렉토리와 파일명을 가진 File객체 생성

if (mFile.getSize() != 0) {

if (!file.exists()) {

전달받은 파일의 크기가0이 아니고, 파일 시스템에 file이 없다면

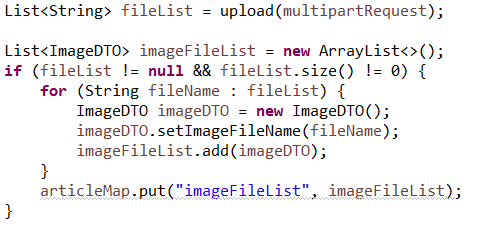


file에 저장한 디렉토리를 생성하고 imagerepo에 저장함 filename 파일을

temp폴더에 originalFileName으로 저장

originalFileName들을 저장한 filList 리턴

다시 addNewArticle로

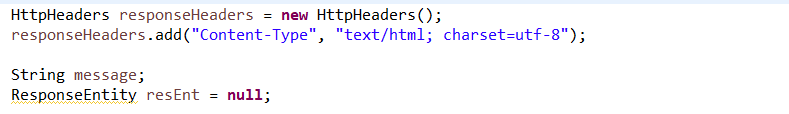


upload(multipartRequest)로 가져옴 originalFileName들을

ImageDTO의 ImageFileName 속성에 저장하고

ImageDTO를 imageFileList에 저장

articleMap에 “imageFileList”의 key값으로 저장

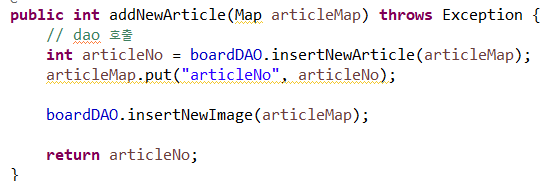


ResponsEntity의 헤더의 담을 responseHeaders 생성

responseHeaders 에 encoding 정보 입력

전달할 String 객체 message와 resEnt 생성





title, content , id 저장된 articleMap으로 boardService.addNewArticle 실행



boardDAO.insertNewArticle로 aritcleNO 구함

aricleMap에 aritcleNo 삽입

boardDAO.insertNewImage(articleMap) 이미지정보 저장

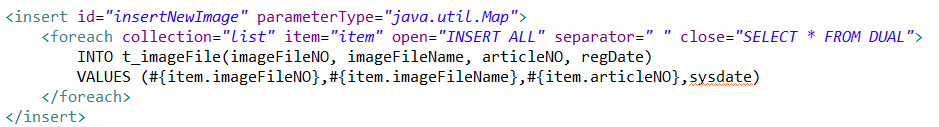


origialFileName속성을 저장한 imageDTO imageFileList를 값으로 지정



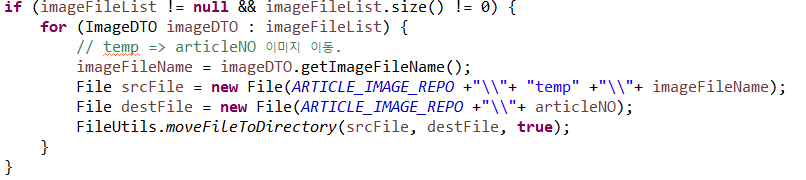
현재 최신 글 번호 가져옴

imageFileList의 imageDTO들에 ++imageNO값과 현재글의 aritcleNO 값 주입

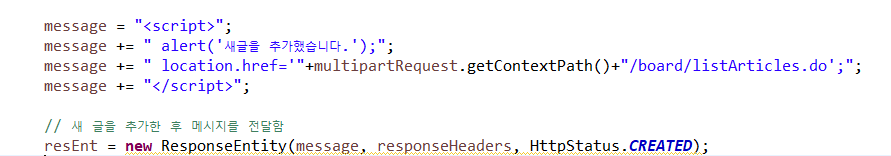


foreach를 통해 값을 순회하여 imageDTO들 전송

articleNo 리턴하고 service.addNewArticle(articleMap) 완료



temp에 저장된 파일을 articleNO의 이름을 가진 desFIile로 이동



message와 헤더 HttpStatus 를 가지 ResEnt를 view에 전달



오류 발생 시 temp에 저장된 파일들 제거하고 resEnt view 전달