[14] FileUpload

1. Pom.xml에 의존추가

<!-- 파일업로드 -->

<dependency>

<groupId>commons-fileupload</groupId>

<artifactId>commons-fileupload</artifactId>

<version>1.3.1</version>

</dependency>

1. Web.xml 한글필터처리부분 추가
2. Servlet-context.xml (웹과 관련된 설정부분 여기다) cf.웹이외의 설정부분은 root-context.xml
   * multipartResolver Bean생성

<beans:bean id=*"multipartResolver"* class=*"org.springframework.web.multipart.commons.CommonsMultipartResolver"*>

<!--

maxUploadSize : 최대 업로드 가능한 바이트 크기

maxInMemorySize : 디스크에 임시파일을 생성하기 전에 메모리에 보관할 수 있는 최대 바이트 크기

defaultEncoding : 요청을 파싱할 때 사용할 캐릭터 인코딩. 기본은 ISO-8859-1

-->

<beans:property name=*"maxUploadSize"* value=*"10485760"*/>

</beans:bean>

1. view

* **file\_input.jsp**

<%@ page language=*"java"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>

<title>Insert title here</title>

</head>

<body>

<div align=*"center"*>

<hr width=*"500"* color=*"green"*>

<h2>파일 upload</h2>

<hr width=*"500"* color=*"green"*>

<form action=*"fileUpload"* method=*"post"* enctype=*"multipart/form-data"*>

<p>첨부파일1 : <input type=*"file"* name=*"file1"*></p>

<p>첨부파일2 : <input type=*"file"* name=*"file2"*></p>

<p>첨부파일3 : <input type=*"file"* name=*"file3"*></p>

<input type=*"submit"* value=*"전송"*>

</form>

</div>

</body>

</html>

* + **fileResult.jsp**

<%@ page language=*"java"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>

<title>Insert title here</title>

</head>

<body>

<h2>전송결과 : ${fileResult}</h2>

파일0 : upload/${filename[0] }<br>

파일1 : upload/${filename[1] }<br>

파일2 : upload/${filename[2] }<br>

<img src=*"upload/*${filename[0] }*"*>

<img src=*"upload/*${filename[1] }*"*>

<img src=*"upload/*${filename[2] }*"*>

</body>

</html>

1. Controller
   * FileController.java

**package** com.ch.ch17;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.multipart.MultipartHttpServletRequest;

**import** org.springframework.web.servlet.ModelAndView;

**import** com.ch.ch17.service.Upload;

@Controller

**public** **class** FileController {

@Autowired

**private** FileUpLoad upload;

@RequestMapping(value="fileInput")

**public** ModelAndView fileInput(){

ModelAndView mav = **new** ModelAndView();

mav.setViewName("fileInput");

**return** mav;

}

@RequestMapping(value="/fileUpload", method=RequestMethod.***POST***)

**public** ModelAndView fileUpload(MultipartHttpServletRequest mRequest){

ModelAndView mav = **new** ModelAndView();

//파일을 서버에 지정된 위치에 저장.

**if**(upload.fileUpload(mRequest, mav)){

mav.addObject("fileResult", "파일업로드 성공");

}**else**{

mav.addObject("fileResult", "파일업로드 실패");

}

mav.setViewName("fileResult");

**return** mav;

}

}

1. Service
   * Upload.java

**package** com.ch.ch17.service;

**import** java.io.File;

**import** java.io.FileInputStream;

**import** java.io.FileOutputStream;

**import** java.io.IOException;

**import** java.util.Iterator;

**import** org.springframework.stereotype.Service;

**import** org.springframework.web.multipart.MultipartFile;

**import** org.springframework.web.multipart.MultipartHttpServletRequest;

**import** org.springframework.web.servlet.ModelAndView;

@Service

**public** **class** FileUpLoad {

**public** **boolean** fileUpload(MultipartHttpServletRequest mRequest, ModelAndView mav){

**boolean** isUpload = **false**;

String uploadPath = mRequest.~~getRealPath~~("upload/");

String savePath = "C:/a01\_prog/SourceS/4\_spring/fileup/src/main/webapp/upload/";

System.***out***.println(uploadPath);

Iterator<String> iterator = mRequest.getFileNames();

String[] saveFileNames = **new** String[3];

**int** i=0;

**while**(iterator.hasNext()){

String uploadFileName = iterator.next();

MultipartFile mFile = mRequest.getFile(uploadFileName);//실제화일 받아옴

String originFileName = mFile.getOriginalFilename();

saveFileNames[i] = originFileName; // 실제저장되는 파일 이름

**if**(saveFileNames[i]!=**null** && !saveFileNames[i].equals("")){

System.***out***.println(i+"번째"+originFileName);

**if**(**new** File(uploadPath+saveFileNames[i]).exists() || **new** File(savePath+saveFileNames[i]).exists()){

saveFileNames[i] = System.*currentTimeMillis*() + "\_" + saveFileNames[i];

}

**try** {

mFile.transferTo(**new** File(uploadPath+saveFileNames[i]));

System.***out***.println("실제 업로드된 파일 : "+uploadPath+saveFileNames[i]);

System.***out***.println("서버에서 복사된 파일 : "+savePath+saveFileNames[i]);

isUpload = copy(uploadPath+saveFileNames[i], savePath+saveFileNames[i]);

isUpload = **true**;

} **catch** (IllegalStateException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}//try

i++;

}//if

}//while

mav.addObject("filenames", saveFileNames);

**return** isUpload;

}

**private** **boolean** copy(String originalFile, String targetFile) {

**boolean** isCopy = **false**;

FileInputStream is = **null**;

FileOutputStream os =**null**;

**try**{

is = **new** FileInputStream(originalFile);

os = **new** FileOutputStream(targetFile);

**byte**[] buff = **new** **byte**[1024\*1024];

**int** nRead;

**while**((nRead = is.read(buff))!=-1){

os.write(buff, 0, nRead);

}

isCopy = **true**;

}**catch** (IOException e) {

System.***out***.println(e.getMessage());

}**finally** {

**try**{

**if**(is!=**null**) is.close();

**if**(os!=**null**) os.close();

}**catch**(IOException e){

System.***out***.println(e.getMessage());

}

}

**return** isCopy;

}

}

1. Servlet-context.xml 에 추가

<resources mapping=*"/upload/\*\*"* location=*"/upload/"* />