## 上传及下载文件 smartupload

### 一、最初的上传文件

1. 创建对象：

SmartUpload upload = **new** SmartUpload();

1. 初始化对象：

upload.initialize(**this**.getServletConfig(), request, response);

1. 原版本不存在，修改乱码的问题（注意，一定要写）

upload.setCharSet("utf-8");

1. 接收上传文件保存到服务器内存中，upload

upload.upload();

1. 从服务器内存保存到硬盘中，减少服务器负担

upload.save("f:/");

html、中写入：

<form action=*"uploadss.do"* method=*"post"* enctype=*"multipart/form-data"*>

<input type=*"file"* name=*"f"*/><br>

<input type=*"submit"* value=*"确定"* name=*"f"*>

</form>

Servlet中：

**public** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

SmartUpload upload = **new** SmartUpload();

// String fname = upload.getRequest().getParameter("f");

upload.initialize(**this**.getServletConfig(), request, response);

upload.setCharSet("utf-8");

**try** {

upload.upload();

upload.save("f:/");

} **catch** (SmartUploadException e) {

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

e.printStackTrace();

}

}

注意的地方：

1. 必须修改乱码 upload.setCharSet("utf-8");不然就会报null，老师加的方法
2. 当html中加入<input type=*"text"* name=*"wu"*><br> 用普通的方法接收不到，由于上传格式是，*multipart/form-data，***显示为null**

必须使用

upload.save("f:/");

String fname = upload.getRequest().getParameter("wu");

**必须要写在文件保存的后面，才能接收成功**

### 二、修改上传多个文件修改文件名：

**public** **void** doPost(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

SmartUpload upload = **new** SmartUpload();

upload.initialize(**this**.getServletConfig(), request, response);

upload.setCharSet("utf-8");

**try** {

upload.upload();

// upload.save("f:/");

Files files = upload.getFiles();

System.*out*.println(files.getSize()); //文件的大小

//System.out.println( files.getCount());文件数量

**if**(files.getSize() < 66666){//

**for**(**int** i =0;i < files.getCount();i++){

File file = files.getFile(i);

String ext = file.getFileExt();//拿到文件的后缀名

String dates = **new** SimpleDateFormat("yyyyMMddHHmmssSSS").format(**new** Date());

Random random = **new** Random();

**int** j = random.nextInt(1000);

String saveurl = dates+j+"."+ext; //保存路径

file.saveAs(saveurl,SmartUpload.*SAVE\_PHYSICAL*);//保存服务器tomcat中

System.*out*.println(saveurl);

}} **catch** (SmartUploadException e) {

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

e.printStackTrace();

}**finally**{

upload = **null**;//使用后为了减少内存，使其为null，更快的被回收

}

}

### 下载文件设置：

**public** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

// 新建一个SmartUpload对象

SmartUpload upload = **new** SmartUpload();

// 初始化

upload.initialize(getServletConfig(), request, response);

// 设定contentDisposition为null以禁止浏览器自动打开文件，

//保证点击链接后是下载文件。若不设定，则下载的文件扩展名为

//doc时，浏览器将自动用word打开它。扩展名为pdf时，浏览器将用acrobat打开。

upload.setAllowedFilesList("jpg,gif,png,jpeg,bmp,xls");// 此处的文件格式可以根据需要自己修改

// 下载文件

**try** {

upload.downloadFile("C:/Users/admin/Downloads/jlkj.xls");

} **catch** (SmartUploadException e) {

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

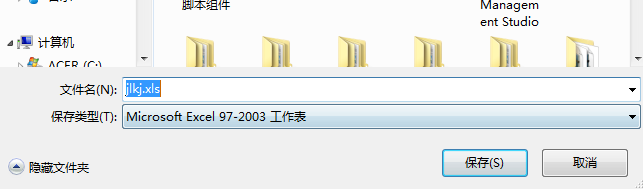
e.printStackTrace();

} **finally**{

upload = **null**;//使用后为了减少内存，使其为null，更快的被回收

}

}



### 主要方法介绍

#### 上传

// 新建一个SmartUpload对象

**SmartUpload su = new SmartUpload();**

// 上传初始化

**su.initialize(pageContext);**

// 限制每个上传文件的最大长度。

**su.setMaxFileSize(10000);**

// 限制总上传数据的长度。

**su.setTotalMaxFileSize(20000);**

// 设定允许上传的文件（通过扩展名限制）,仅允许doc,txt文件。 **su.setAllowedFilesList("doc,txt");**

// 设定禁止上传的文件（通过扩展名限制）,禁止上传带有exe,bat, //jsp,htm,html扩展名的文件和没有扩展名的文件。 **su.setDeniedFilesList("exe,bat,jsp,htm,html,,");**

// 上传文件 su.upload(); // 将上传文件全部保存到指定目录

**int count = su.save("/upload");**

#### 下载

// 新建一个SmartUpload对象

**SmartUpload su = new SmartUpload();**

// 初始化

**su.initialize(pageContext);**

// 设定contentDisposition为null以禁止浏览器自动打开文件，

//保证点击链接后是下载文件。若不设定，则下载的文件扩展名为

//doc时，浏览器将自动用word打开它。扩展名为pdf时，浏览器将用acrobat打开。

**su.setContentDisposition(null);**

// 下载文件

**su.downloadFile("/upload/test.doc");**

### 应用实例： 用户只能上传图片格式的文件实例

uploadimage.jsp

<%@ page contentType="text/html;charset=gb2312" language="java" import="java.io.\*,java.awt.Image,java.awt.image.\*,com.sun.image.codec.jpeg.\*,java.sql.\*,com.jspsmart.upload.\*,java.util.\*"%>

<% SmartUpload mySmartUpload =new SmartUpload();

long file\_size\_max=4000000;

String fileName2="",ext="",testvar="";

String url="upload/";

//应保证在根目录中有此目录的存在（也就是说需要自己建立相应的文件夹）

//初始化

mySmartUpload.initialize(pageContext); //只允许上载此类文件

try { mySmartUpload.setAllowedFilesList("jpg,gif");

//此处的文件格式可以根据需要自己修改 //上载文件

mySmartUpload.upload();

} catch (Exception e){ %> <SCRIPT language=javascript> alert("只允许上传.jpg和.gif类型图片文件");

window.location='upload.htm'; </script> <% }

try{

com.jspsmart.upload.File myFile = mySmartUpload.getFiles().getFile(0);

if (myFile.isMissing()){

%><SCRIPT language=javascript> alert("请先选择要上传的文件"); window.location='upload.htm';

</script> <%}

else{ //String myFileName=myFile.getFileName();

//取得上载的文件的文件名

ext= myFile.getFileExt(); //取得后缀名

int file\_size=myFile.getSize(); //取得文件的大小

String saveurl=""; if(file\_size<file\_size\_max){

//更改文件名，取得当前上传时间的毫秒数值

### 四 SmartUpload类这个类完成上传下载工作。

#### A．上传与下载共用的方法：

只有一个：initialize。

作用：执行上传下载的初始化工作，必须第一个执行。

原型：有多个，主要使用下面这个：

public final void initialize(javax.servlet.jsp.PageContext pageContext) 其中，pageContext为JSP页面内置对象（页面上下文）。

B．上传文件使用的方法：

##### 1、upload

作用：上传文件数据。对于上传操作，第一步执行initialize方法，第二步就要执行这个方法。

原型：public void upload()

##### 2、save

作用：将全部上传文件保存到指定目录下，并返回保存的文件个数。

原型：public int save(String destPathName) 和public int save(String destPathName,int option) 其中，destPathName为文件保存目录，option为保存选项，它有三个值，分别是**SAVE\_PHYSICAL,SAVE\_VIRTUAL和SAVE\_AUTO**。（同File类的saveAs方法的选项之值类似）SAVE\_PHYSICAL指示组件将文件保存到以操作系统根目录为文件根目录的目录下（tomcat中bin），SAVE\_VIRTUAL指示组件将文件保存到以Web应用程序根目录为文件根目录的目录下，而SAVE\_AUTO则表示由组件自动选择。

注：save(destPathName)作用等同于save(destPathName,SAVE\_AUTO)。

##### 3、getSize

作用：取上传文件数据的总长度

原型：public int getSize()

##### 4、getFiles

作用：取全部上传文件，以Files对象形式返回，可以利用Files类的操作方法来获得上传文件的数目等信息。

原型：public Files getFiles()

##### 5、getRequest

作用：取得Request对象，以便由此对象获得上传表单参数之值。

原型：public Request getRequest()

##### 6、setAllowedFilesList

作用：设定允许上传带有指定扩展名的文件，当上传过程中有文件名不允许时，组件将抛出异常。

原型：public void setAllowedFilesList(String allowedFilesList) 其中，allowedFilesList为允许上传的文件扩展名列表，各个扩展名之间以逗号分隔。如果想允许上传那些没有扩展名的文件，可以用两个逗号表示。

例如：

setAllowedFilesList("doc,txt,,")将允许上传带doc和txt扩展名的文件以及没有扩展名的文件。

##### 7、setDeniedFilesList

作用：用于限制上传那些带有指定扩展名的文件。若有文件扩展名被限制，则上传时组件将抛出异常。

原型：public void setDeniedFilesList(String deniedFilesList)

其中，deniedFilesList为禁止上传的文件扩展名列表，各个扩展名之间以逗号分隔。如果想禁止上传那些没有扩展名的文件，可以用两个逗号来表示。

例如：setDeniedFilesList("exe,bat,,")将禁止上传带exe和bat扩展名的文件以及没有扩展名的文件。

##### 8、setMaxFileSize

作用：设定每个文件允许上传的最大长度。

原型：public void setMaxFileSize(long maxFileSize)

其中，maxFileSize为为每个文件允许上传的最大长度，当文件超出此长度时，将不被上传。

##### 9、setTotalMaxFileSize

作用：设定允许上传的文件的总长度，用于限制一次性上传的数据量大小。

原型：public void setTotalMaxFileSize(long totalMaxFileSize)

其中，totalMaxFileSize为允许上传的文件的总长度。

C．下载文件常用的方法

1、setContentDisposition

作用：将数据追加到MIME文件头的CONTENT-DISPOSITION域。fileUpandDownload组件会在返回下载的信息时自动填写MIME文件头的CONTENT-DISPOSITION域，如果用户需要添加额外信息，请用此方法。

原型：public void setContentDisposition(String contentDisposition) 其中，contentDisposition为要添加的数据。

如果contentDisposition为null，则组件将自动添加"attachment;"，以表明将下载的文件作为附件，结果是IE浏览器将会提示另存文件，而不是自动打开这个文件（IE浏览器一般根据下载的文件扩展名决定执行什么操作，扩展名为doc的将用word程序打开，扩展名为pdf的将用acrobat程序打开，等等）。

##### 2、downloadFile

作用：下载文件。

原型：共有以下三个原型可用，第一个最常用，后两个用于特殊情况下的文件下载（如更改内容类型，更改另存的文件名）。 ① public void downloadFile(String sourceFilePathName) 其中，sourceFilePathName为要下载的文件名（带目录的文件全名） ② public void downloadFile(String sourceFilePathName,String contentType) 其中，sourceFilePathName为要下载的文件名（带目录的文件全名）,contentType为内容类型（MIME格式的文件类型信息，可被浏览器识别）。 ③ public void downloadFile(String sourceFilePathName,String contentType,String destFileName)

其中，sourceFilePathName为要下载的文件名（带目录的文件全名）,contentType为内容类型（MIME格式的文件类型信息，可被浏览器识别）,destFileName为下载后默认的另存文件名。

三、文件上传篇

一 表单要求

对于上传文件的FORM表单，有两个要求：

1、METHOD应用POST，即METHOD="POST"。

2、增加属性：ENCTYPE="multipart/form-data" 下面是一个用于上传文件的FORM表单的例子：

<FORM METHOD="POST" ENCTYPE="multipart/form-data" ACTION="/fileUpandDownload/upload.jsp">

<INPUT TYPE="FILE" NAME="MYFILE">

<INPUT TYPE="SUBMIT"> </FORM>

二 上传的例子

1、上传页面upload.html 本页面提供表单，让用户选择要上传的文件，点击"上传"按钮执行上传操作。 页面源码如下：

<!-- 文件名：upload.html --> <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">