1. 文件上传

Struts提交的文件上传组件来上传文件

[前台]：

1. 提交方式为POST
2. 表单类型 enctype=multipart/form-data
3. <input type=”file”/>

[后台]：

Apache提供的文件上传组件FileUpload组件

核心类：

FileItemFactory

ServletFileUpload servlet中文件上传的核心类

FileItem 主要封装了上传的表单文件项的信息

1. Struts文件上传

[1] 前台页面

|  |
| --- |
| <**form action="${**pageContext.request.contextPath**}/uploadAction.action" method="post" enctype="multipart/form-data"**>  请选择文件：<**input type="file" name="file1"**/><**br**/>  <**input type="submit" value="上传文件"**/> |

[2]编写Action

|  |
| --- |
| **public class** UploadAction **extends** ActionSupport {  *//对应表单的名称<input type="file" name="file1"/>* **private** File **file1**;  **private** String **file1FileName**;*//固定写法 file1+FileName，获取文件名称* **private** String **file1ContentType**;*//固定写法 file1+ContentType，获取文件的类型* **public void** setFile1(File file1) {  **this**.**file1** = file1;  }  **public void** setFile1FileName(String file1FileName) {  **this**.**file1FileName** = file1FileName;  }  **public void** setFile1ContentType(String file1ContentType) {  **this**.**file1ContentType** = file1ContentType;  }  @Override  **public** String execute() **throws** Exception {  String path=ServletActionContext.*getServletContext*().getRealPath(**"/upload"**);*//得到上传的目录路径*  File destfile=**new** File(path,**file1FileName**);*//创建目标文件对象*  FileUtils.*copyFile*(**file1**,destfile);*//将上传的文件拷贝到目标文件中*  **return "success"**;  } } |

Struts文件上传大小限制：

Struts默认支持上传文件的大小为2M， 【struts2-core.jar!\org\apache\struts2\default.properties】

|  |
| --- |
| *<!-- 修改上传文件的最大值为50M-->* <**constant name="struts.multipart.maxSize" value="52428800"**/> |

文件上传出现错误时，Struts将会返回input视图，则需要在struts.xml中配置错误页面

|  |
| --- |
| <**result name="input"**>/error.jsp</**result**> |

Struts文件上传类型限制：

需要在struts.xml中配置：

|  |
| --- |
| 限制文件上传的类型：  <**interceptor-ref name="defaultStack"**>  <**param name="fileUpload.allowedExtensions"**>txt,jpeg</**param**> </**interceptor-ref**> |

1. Struts文件下载

**1、**通过response对象向浏览器写入字节流数据，设置下载响应头

**2、**struts文件下载

**步骤：**

**[1]:列出需要下载的文件**

**[2]:在Struts.xml配置文件中配置参数**

**[3]:后台处理相关方法**

|  |
| --- |
| *//1、列出需要下载的文件* **public** String getFilename() **throws** Exception {  *//得到目录所有文件的路径 //String path= ServletActionContext.getServletContext().getRealPath("/upload");  //得到指定文件的路径* String path= ServletActionContext.*getServletContext*().getRealPath(**"/upload/readme.docx"**); File file1=**new** File(path);  *//得到所有文件的名称 //String[] fileNames=file1.list();  //得到指定文件的名称* String fileNames=file1.getName(); ActionContext ac=ActionContext.*getContext*(); Map<String,Object> request= (Map<String, Object>)ac.get(**"request"**); *//request.put("fileNames",fileNames[0]);* request.put(**"fileNames"**,fileNames); **return "list"**;  } |

|  |
| --- |
| *//获取下载文件的文件名* **private** String **fileName**; **public void** setFileName(String fileName) {  *//处理传入的参数中文乱码问题* **try** {  fileName=**new** String(fileName.getBytes(**"ISO8859-1"**),**"UTF-8"**);  } **catch** (UnsupportedEncodingException e) {  **throw new** RuntimeException(e);  }  **this**.**fileName** = fileName; }  */\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*文件下载\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/* **public** String down() **throws** Exception {  **return "download"**; } *//返回文件流的属性* **public** InputStream getAttrInputStream() {  **return** ServletActionContext.*getServletContext*().getResourceAsStream(**"/upload/"**+**fileName**); } *//获取浏览器显示的文件名* **public** String getDownFileName() {  *//进行中文编码* **try** {  **fileName**= URLDecoder.*decode*(**fileName**,**"UTF-8"**);  } **catch** (UnsupportedEncodingException e) {  **throw new** RuntimeException(e);  }  **return fileName**; }  **Struts配置文件：**  <**result name="download" type="stream"**>  *<!--允许下载的文件类型，（application/octet-stream指定为所有的二进制文件类型）-->* <**param name="contentType"**>application/octet-stream</**param**>  *<!--返回Action中流的属性，即是（getAttrInputStream()）-->* <**param name="inputName"**>attrInputStream</**param**>  *<!--下载头，包括浏览器显示的文件名-->* <**param name="contentDisposition"**>attachment;filename=${downFileName}</**param**>  *<!--缓冲区的大小-->* <**param name="bufferSize"**>1024</**param**> </**result**>  **前台：**  <**a href="${**pageContext.request.contextPath**}/downloadAction\_down.action?fileName=${**fileName**}"**>下载</**a**>  显示所有文件：  **<table border="1" align="center">**  **<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>**  **<c:forEach var="fileName" items="${fileNames}" varStatus="vs">**  **<tr>**  **<td>${vs.count }</td>**  **<td>${fileName }</td>**  **<td>**  **<!-- 构建一个url -->**  **<c:url var="url" value="down\_down">**  **<c:param name="fileName" value="${fileName}"></c:param>**  **</c:url>**    **<a href="${url }">下载</a>**  **</td>**  **</tr>**  **</c:forEach>**  **</table>** |

Struts图片的上传：

|  |
| --- |
| *//将上传的图片保存到session中* ActionContext ac=ActionContext.*getContext*(); Map<String,Object> map= (Map<String, Object>) ac.get(**"session"**); map.put(**"img"**,**imgfileFileName**);  前台：<**img src="${**pageContext.request.contextPath**}/images/${**img**}"**/> |