

## 文件上传的三个必要条件：

- 1、请求的方式必须是POST
- 2、表单的enctype取值必须是： multipart/form-data
- 3、必须在表单中提供一个文件选择域： input type="file"

## 上传原理：

请求参数用getParameter取不出来

原因是：请求参数的体现形式变了，原来能取出来是因为体现形式是键值对。现在不是了，现在是多部分表单数据。用分隔符分隔的。

用request的getInputStream()可以取到请求正文。

借助第三方组件实现文件上传： commons-fileupload

核心类： ServletFileUpload

创建核心类工厂： DiskFileItemFactory

## 实现步骤：

- 1、判断表单是否支持文件上传

```
ServletFileUpload.isMutilPartContent(request);
```

- 2、创建工厂

```
DiskFileItemFactory factory = new DiskFileItemFactory();
```

- 3、根据工厂创建核心类

```
ServletFileUpload sfu = new ServletFileUpload(factory);
```

- 4、使用工具类解析 request,得到一个由FileItem组成的List

```
List<FileItem> items = sfu.parseRequest(request);
```

- 5、遍历List，取出每个FileItem

```
for(FileItem item : items){
```

- 6、判断FileItem，用isFormField

```
if(item.isFormField()){
```

```
    返回true 普通字段：封装到javabean中
```

```

String fieldName = item.getFieldName();

String fieldValue = item.getString("utf-8");

}else{
返回false 文件字段，写到服务器

String fileName = item.getName();

File file = new File(path+"/"+fileName);

item.write(file);

item.delete();

}

}

```

Struts2的文件上传使用：

[SaleVisitAction.java](#)

//1.根据id查询拜访详情对象

```
SaleVisit dbSaleVisit = saleVisitService.findSaleVisitById(visit.getVisitId());
```

//2.获取文件保存的磁盘目录

```
ServletContext application = ServletActionContext.getServletContext();
```

```
String basePath = application.getRealPath("/uploads");
```

//3.根据查询出来对象的文件名和日期目录再加载保存磁盘目录创建一个字节输入流

//获取文件名

```
String GUIDFileName = dbSaleVisit.getVisitFileName();
```

```
InputStream inputStream = new
```

```
FileInputStream(basePath+File.separator+dbSaleVisit.getVisitFilePath()+File.separator+GUIDFileName);
```

//4.获取response对象

```
HttpServletResponse response = ServletActionContext.getResponse();
```

```
ServletOutputStream sos = response.getOutputStream();
```

//5.设置两个消息头： a.响应正文的MIME类型 b.以下载的方式打开

```
response.setHeader("Content-Type", "application/octet-stream");//bin 二进制
```

//去掉文件名前面的GUID部分

```

String fileName =
GUIDFileName.substring(GUIDFileName.indexOf("_")+1);//EPYX2IFYD8OR0VARX9FUF6RU4FC9OKL_
客户拜访详情.docx

response.setHeader("Content-Disposition",
"attachment;filename="+java.net.URLEncoder.encode(fileName,"UTF-8"));

//6.流的操作

int len = 0;

byte[] by = new byte[1024];

while((len = inputStream.read(by)) != -1){

    sos.write(by,0,len);

}

inputStream.close();

```

时间UI控件 使用：

1、导入jquery:

<!-- 日期插件，使用jquery -->

<script type="text/javascript" src="{pageContext.request.contextPath}/jquery/jquery-1.4.2.js"></script>

<link rel="stylesheet" href="{pageContext.request.contextPath}/jquery/jquery.datepick.css"  
type="text/css">

<script type="text/javascript" src="{pageContext.request.contextPath}/jquery/jquery.datepick.js">  
</script>

<script type="text/javascript"

src="{pageContext.request.contextPath}/jquery/jquery.datepick-zh-CN.js"></script>

1、定义时间选择

<td>拜访时间: </td>

<td>

<s:textfield name="visitTime" cssClass="textbox" id="visitTime" readonly="true" cssStyle="WIDTH:  
180px" maxLength="50"/>

</td>

<td>下次拜访时间: </td>

<td>

<s:textfield name="visitNextTime" cssClass="textbox" id="visitNextTime" readonly="true"

```
cssStyle="WIDTH: 180px" maxLength="50"/>
```

```
</td>
```

## 2、使用jquery的UI

```
<script type="text/javascript">
```

```
$(document).ready(function(){
```

```
//使用class属性处理 'yy-mm-dd' 设置格式"yyyy/mm/dd"
```

```
$('#visitTime').datepicker({dateFormat: 'yy-mm-dd'});
```

```
$('#visitNextTime').datepicker({dateFormat: 'yy-mm-dd'});
```

```
});
```

```
</script>
```