

Servlet ↗

基础

■ 开发servlet (/k/servlet/servlet-eclipse/558.html)

获取参数 (/k/servlet/servlet-paramter/547.html)

返回响应 (/k/servlet/servlet-response/548.html)

调用流程 (/k/servlet/servlet-flow/775.html)

service() (/k/servlet/servlet-service/549.html)

中文问题 (/k/servlet/servlet-gbk/554.html)

生命周期 (/k/servlet/servlet-lifecycle/550.html)

跳转 (/k/servlet/servlet-jump/551.html)

自启动 (/k/servlet/servlet-startup/553.html)

request常见方法 (/k/servlet/servlet-request/555.html)

response 用法 (/k/servlet/servlet-response/556.html)

上传文件 (/k/servlet/servlet-upload/587.html)

动态Web项目

CRUD

JSON

J2EE (/STAGE/13.HTML) / SERVLET (/MODULE/71.HTML) / SERVLET 上传文件逐步演示

← response 用法 (/k/servlet/servlet-response/556.html) 创建 → (/k/servlet/servlet-dynamic-web-project/1306.html)

工具版本兼容问题 (/k/helloworld/helloworld-version/1718.html)

相关下载

文件名	文件大小
commons-fileupload-1.2.2.jar (/frontdownload?bean.id=293)	58k
commons-io-1.4.jar (/frontdownload?bean.id=294)	106k
j2ee.rar (/frontdownload?bean.id=1471)	169k

使用站长秘制下载工具 (/k/1381)

步骤 1: 先运行, 看到效果, 再学习

步骤 2: 模仿和排错

步骤 3: 首先准备上传页面 upload.html

步骤 4: 接着准备 UploadPhotoServlet

步骤 5: 配置web.xml

步骤 6: 复制jar包

步骤 7: 在项目中导入jar包

步骤 8: 上传测试

步骤 9: 如何处理其他非File字段

步骤 1: 先运行, 看到效果, 再学习

老规矩, 先下载右上角的可运行项目, 配置运行起来, 确认可用之后, 再学习做了哪些步骤以达到这样的效果。

步骤 2: 模仿和排错

在确保可运行项目能够正确无误地运行之后, 再严格照着教程的步骤, 对代码模仿一遍。模仿过程难免代码有出入, 导致无法得到期望的运行结果, 此时此刻通过比较 正确答案 (可运行项目) 和自己的代码, 来定位问题所在。采用这种方式, 学习有效果, 排错有效率, 可以较为明显地提升学习速度, 跨过学习路上的各个槛。推荐使用diffmerge软件, 进行文件夹比较。把你自己做的项目文件夹, 和我的可运行项目文件夹进行比较。这个软件很牛逼的, 可以知道文件夹里哪两个文件不对, 并且很明显的标记出来 这里提供了绿色安装和使用教程: diffmerge 下载和使用教程 (/k/18/18-1583/1583.html)

步骤 3: 首先准备上传页面 upload.html

上传页面有两点需要注意

1. form 的 method必须是post 的, get不能上传文件。还需要加上 enctype="multipart/form-data" 表示提交的数据是二进制文件

```
<form action="uploadPhoto" method="post" enctype="multipart/form-data">
```

2. 需要提供 **type="file"** 的字段进行上传

[复制代码](#)[代码比较](#)

```
1 <!DOCTYPE html>
2
3 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4 <form action="uploadPhoto" method="post" enctype="multipart/form-data">
5 英雄名称:<input type="text" name="heroName" /> <br>
6 上传头像 : <input type="file" name="filepath" /> <br>
7 <input type="submit" value="上传">
8 </form>
```

#### 步骤 4: 接着准备 UploadPhotoServlet

[折](#)[顶](#)

在UploadPhotoServlet进行上传的功能开发。

1. 需要用到两个第三方的jar包, **commons-io-1.4.jar** 和 **commons-fileupload-1.2.2.jar**。在右侧下载, 并放在WEB-INF/lib下

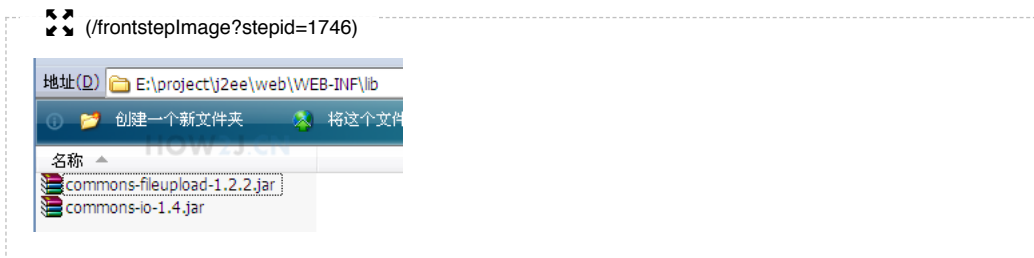
2. 前部分代码是固定写法, 用来做一些准备工作。直到遍历出Item, 一个Item就是对应一个浏览器提交的数据, 通过item.getInputStream可以打开浏览器上传的文件的输入流。

3. 客户提交的文件名有可能是一样的, 所以在服务端保存文件的时候, 不能使用客户提交的文件名。这里使用的是一种粗糙的解决文件名重复的办法, 即使用时间戳。

4. 读取输入流中的数据, 保存在服务端的目录下 e:/project/j2ee/web/uploaded, 这个目录是通过getRealPath获取到的。如果项目部署在其他地方, 那么会自动做相应的变化。

注: 为什么要放这里? 因为后续网页上显示的时候是通过http://127.0.0.1/uploaded/xxx.jpg 路径来获取图片。

5. 根据临时生成的文件名, 创建一个html img元素, 然后通过response返回浏览器



代码行数较多, 请点击查看 </>

[复制代码](#)[代码比较](#)

#### 步骤 5: 配置web.xml

[折](#)[顶](#)

配置UploadPhotoServlet

[复制代码](#)[代码比较](#)

```
1 <servlet>
2   <servlet-name>UploadPhotoServlet</servlet-name>
3   <servlet-class>UploadPhotoServlet</servlet-class>
4 </servlet>
5
6 <servlet-mapping>
7   <servlet-name>UploadPhotoServlet</servlet-name>
8   <url-pattern>/uploadPhoto</url-pattern>
9 </servlet-mapping>
```

#### 步骤 6: 复制jar包

[折](#)[顶](#)

再强调一遍：把右侧的两个jar包commons-io-1.4.jar， commons-fileupload-1.2.2.jar 复制到WEB-INF/lib 目录下



步骤 7： 在项目中导入jar包

折 顶

把jar包导入到项目中，导包办法：右键 project->properties->java build path->libraries->add external jars

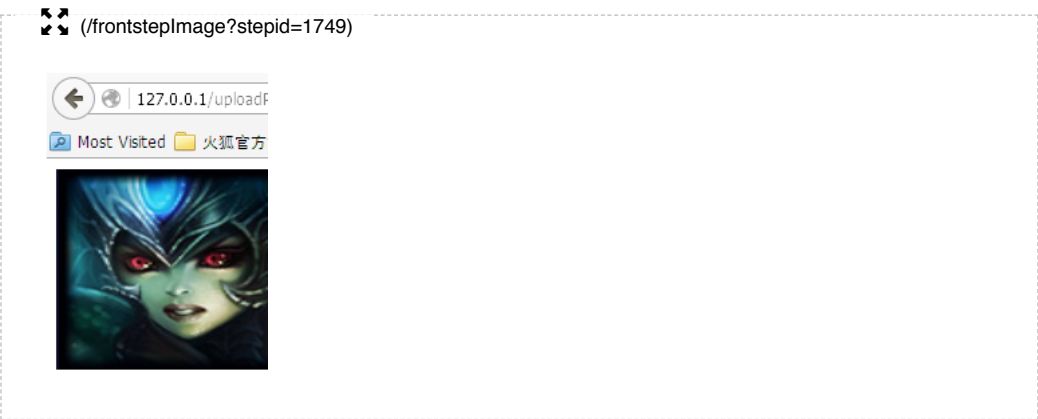
步骤 8： 上传测试

折 顶

重启tomcat, 打开页面进行上传

http://127.0.0.1/upload.html

选中某个图片点击上传，就可以看到上传的图片了



步骤 9： 如何处理其他非File字段

折 顶

因为浏览器指定了以二进制的形式提交数据，那么就不能通过常规的手段获取非File字段：

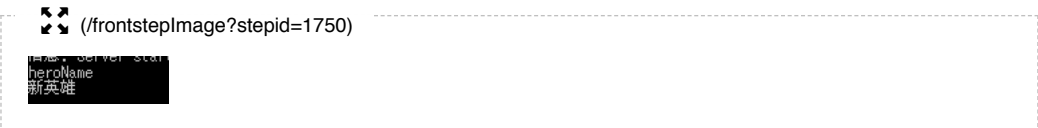
```
request.getParameter("heroName")
```

在遍历Item时(Item即对应浏览器提交的字段)，可以通过

```
item.isFormField
```

来判断是否是常规字段还是提交的文件。当item.isFormField返回true的时候，就表示是常规字段。

然后通过item.getFieldName()和item.getString()就知道分别是哪个字段，以及字段的值了。



代码行数较多, 请点击查看 `</>`

复制代码

代码比较

← response 用法 (/k/servlet/servlet-response/556.html)

创建 → (/k/servlet/servlet-dynamic-web-project/1306.html)

HOW2J公众号, 关注后实时获知布最新的教程和优惠活动, 谢谢。



问答区域

2018-09-06 关于 **getServletContext(报错)**

关于 **J2EE-Servlet-上传文件** (/k/servlet/servlet-upload/587.html) 的提问

hello  
world

问题

截图

代码

异常

代码复制,只有getServletContext()报错

添加答案

2018-08-08 为什么多了一个img标签啊

2018-07-31 配置了之后就重启服务器出现错误

2018-07-11 关于request.getServletContext().getRealPath("uploaded")的疑问

2018-07-11 如何同时上传多张图片

2018-06-20 idea中使用request.getServletContext().getRealPath("uploaded")出现问  
题?

2018-05-27 ① **item.write(文件名);**



2018-05-08 ① **关于src='uploaded/%s'**



2018-05-04 ① **放到lib下还是闪退**



2018-04-19 ① **获取文件夹路径**



2018-04-15 ① **请问两份代码区别在哪?**



2018-04-11 ① **不明白为什么jar包不能放在别的目录下,在别的目录下启动tomcat会报错**



2018-03-01 ① **原文件名去哪了? 怎么取到?**



2018-02-09 ① **复制进去不好用, 不知道错误在哪里**



2018-01-07 ① **另一种实现方法**



2017-09-16 ① **复制文件**



2017-09-04 ① **真的是不能显示图片**



2017-08-18 ① **上传图片 jar包中的类 加载失败。**



2017-08-10 ① **响应页面没有显示图片**



2017-08-06 ① **请问头像的文件夹如何改成相对路径**



2017-08-02 ① **上传文件路径问题**



2017-08-01 ① **直接拉进来项目可以看到效果, 但是一打开src里都是错误, 最后我绑定了jre system library后, 还有个pw.format报错**



2017-07-13 ① **一启动tomcat, 直接跳ClassNotFoundException异常, 然后闪退**



2017-07-12 ① **看不到图片内容**



2017-05-24 ① **为什么我上传图片后在网页中没看见图片**



2017-05-23 ② **import org.apache.commons.fileupload.FileItem;**



2017-05-22 ② **用commons的fileupload包的问题**



2017-05-11 ② **关于web.xml文件**



2017-04-20 ② **图片状态：404 Not Found**



2017-04-14 ② **在Chrome浏览器中输出的是html 没有进行解析 在ie是正常的**



2017-04-08 ② **不显示图片问题**



2017-02-07 ② **servlet3.0后好像不用这么麻烦了？**



2016-12-22 ② **J2EE-Servlet-上传文件**



2016-09-18 ② **TOMCAT闪退**



2016-07-28 ② **上传测试,不显示图片**



2016-07-19 ② **无法正常显示图片**



2016-07-04 ② **提交后没结果**



提问 提问之前请登陆 (/frontloginPage)

