

传统的文件上传

一、文件上传

1、要实现Web开发的上传功能，通常需要完成两步操作：

- 一是在Web页面中添加上传输入项；
- 二是在Servlet中读取上传文件的数据，并保存到本地硬盘中。

上传大多数情况是通过表单的形式提交给服务器，使用<input type="file">标签。

2、上传组件有两种

- FileUpload【操作比较复杂】
- SamrtUpload【操作比较简单】

这里主要讲FileUpload

二、FileUpload注意事项：

1、使用<input type="file">标签需要注意以下两点：

- i. 必须设置name属性，不然浏览器不会发送上传文件的数据。
- ii. 必须将method属性设置为post，ectype属性设置为“multipart/form-data”类型。
- iii.

2、要使用FileUpload组件，就需要导入两个jar包

- commons-io
- Commons-fileupload

3、开发步骤

- **创建解析器工厂对象【DiskFileItemFactory】**
- **通过解析器工厂创建解析器【ServletFileUpload】**
- **调用解析器方法解析request对象，得到所有上传的内容【list】**
- **遍历list，判断每个对象是否是上传文件**
 - **如果是普通表单字段，得到字段名和字段值**
 - **如果是上传文件，调用InputStream方法得到输入流，读取上传的数据**

由于在Servlet直接读取上传数据，并且解析出相应文件数据是一项非常麻烦的工作。为了方便处理上传数据，Apache组织提供了一个开源组件Commons-FileUpload。该组件可以将“multipart/form-data”类型请求的各种表单域解析出来，并实现一个或多个文件上传，同时也可以限制上传文件的大小等内容。其性能十分优异，使用极其简单。

三、实现步骤：

1、导入jar包

```
<dependency>
  <groupId>commons-fileupload</groupId>
  <artifactId>commons-fileupload</artifactId>
  <version>1.3.1</version>
</dependency>
<dependency>
  <groupId>commons-io</groupId>
  <artifactId>commons-io</artifactId>
  <version>2.4</version>
</dependency>
```

2、设置表单页面

```
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<html>
<head>
  <title>Title</title>
</head>
<body>
  <h3>传统文件上传</h3>
```

```
<form action="user/fileupload1" method="post" enctype="multipart/form-  
data">
```

```
    选择文件: <input type="file" name="upload" /> <br/>
```

```
    <input type="submit" value="上传" />
```

```
</form>
```

```
</body>
```

```
</html>
```

3、配置上传代码

```
@Controller
```

```
@RequestMapping("/user")
```

```
public class UserController {
```

```
    @RequestMapping("/fileupload1")
```

```
    public String fileupload1(HttpServletRequest request) throws Exception {
```

```
        System.out.println("文件上传...");
```

```
        // 使用fileupload组件完成文件上传
```

```
        // 上传的位置
```

```
        String path =
```

```
request.getSession().getServletContext().getRealPath("/uploads/");
```

```
        // 判断, 该路径是否存在
```

```
        File file = new File(path);
```

```
        if(!file.exists()){
```

```
            // 创建该文件夹
```

```
            file.mkdirs();
```

```
        }
```

```
        // 解析request对象, 获取上传文件项
```

```
        //1.得到解析器工厂
```

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

```
        //2.得到解析器
```

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

```
        // //为上传表单, 则调用解析器解析request
```

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

```
        // 遍历
```

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

```
            // 进行判断, 当前item对象是否是上传文件项
```

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

```
                // 说明普通表单向
```

```
            }else{
```

```
                // 说明上传文件项
```

```
                // 获取上传文件的名称
```

```
                String filename = item.getName();
```

```
        // 把文件的名称设置唯一值, uuid
        // String uuid = UUID.randomUUID().toString().replace("-", "");
        //filename = uuid+"_"+filename;
    // 完成文件上传
    item.write(new File(path,filename));
    // 删除临时文件
    item.delete();
    }
}

return "success";
}
}
```

文件所在位置：项目路径\target\SpringMVCTest03\uploads