

## 文件上传

- JSP
  - input的type设置为file
  - form表单的method设置为post
  - form表单的enctype设置为multipart/form-data（二进制数据）
- Servlet
  - 接收数据不能调用request.getParameter()获取，必须通过二进制流来获取数据。

### 使用工具完成上传

- 导入commons-io, commons-fileupload。
- 通过工具类的相关方法完成上传。

工具的基本思想：commons-fileupload可以解析request，得到一个List

FileItem是工具自己封装的一个类，该类的每一个实例化对象对应JSP页面form表单中的一个input标签的值（除了提交，重置按钮之外）。

拿到List之后，遍历集合，取出集合中的每一个FileItem对象，再调用对象的相关方法完成上传操作。

### JSP

```
<%--
    Created by IntelliJ IDEA.
    User: southwind
    Date: 2018/12/3
    Time: 8:12 PM
    To change this template use File | Settings | File Templates.
--%>
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<html>
<head>
    <title>Title</title>
</head>
<body>
    <form action="upload.do" method="post" enctype="multipart/form-data">
        <c:if test="${sessionScope.message != null}">
            ${sessionScope.message}<br/>
        </c:if>
        <input type="text" name="desc"/>
        <input type="file" name="file"/><br/>
        <input type="submit" value="上传"/>
    </form>
```

```
</body>
</html>
```

## Servlet

```
//1.创建DiskFileItemFactory
DiskFileItemFactory factory = new DiskFileItemFactory();
//2.创建文件解析器
ServletFileUpload upload = new ServletFileUpload(factory);
//设置上传文件的上限, 以byte为单位, 不能超过2M
//1byte*1024 = 1KB * 1024 = 1MB * 1024 = 1GB * 1024 = 1TB
upload.setSizeMax(1024*1024*2);
//3.获取List<FileItem>
try {
    List<FileItem> list = upload.parseRequest(req);
    for (FileItem fileItem:list){
        //4.判断FileItem的类型, 是否为表单域 (文本输入)
        //如果是表单域, 表示该值是一段文本, 打印输出
        //如果不是表单域, 表示该值是一个文件, 通过IO完成上传
        if(fileItem.isFormField()){
            String name = fileItem.getFieldName();
            //处理中文乱码
            String value = fileItem.getString("UTF-8");
            System.out.println(name+": "+value);
        }else{
            String path = req.getServletContext().getRealPath("file");
            OutputStream outputStream = new
FileOutputStream(path+"/"+fileItem.getName());
            InputStream inputStream = fileItem.getInputStream();
            int temp = 0;
            while((temp = inputStream.read())!=-1){
                outputStream.write(temp);
            }
            outputStream.close();
            inputStream.close();
        }
    }
} catch (FileUploadException e) {
    String message = "上传的文件大小不能超过2M, 请重新选择上传";
    req.getSession().setAttribute("message",message);
    resp.sendRedirect("upload.jsp");
}
```

## 文件下载

原理：当客户端的一个下载请求发送到服务端之后，首先服务端中的Java程序通过输入流读取在服务端本地存储的目标文件，再通过输出流将文件的二进制数据流写入到响应（response）中，通过response传给了客户端，客户端进行下载即可，使用客户端浏览器自带的下载工具来完成下载。

## JSP

```
<%--
    Created by IntelliJ IDEA.
    User: southwind
    Date: 2018/12/3
    Time: 9:46 PM
    To change this template use File | Settings | File Templates.
--%>
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<html>
<head>
    <title>Title</title>
</head>
<body>
    <a href="download.do?fileName=2.png">2.png</a><br/>
    <a href="download.do?fileName=test.txt">test.txt</a><br/>
    <a href="download.do?fileName=文件上传.md">文件上传.md</a>
</body>
</html>
```

## Servlet

```
resp.setContentType("application/x-msdownload");
String fileName = req.getParameter("fileName");
resp.setHeader("Content-Disposition", "attachment;fileName="+
    URLEncoder.encode(fileName, "UTF-8"));
String path = req.getServletContext().getRealPath("file");
InputStream inputStream = new FileInputStream(path+"/"+fileName);
OutputStream outputStream = resp.getOutputStream();
int temp = 0;
while ((temp = inputStream.read()) != -1){
    outputStream.write(temp);
}
outputStream.close();
inputStream.close();
```