# 文件上传

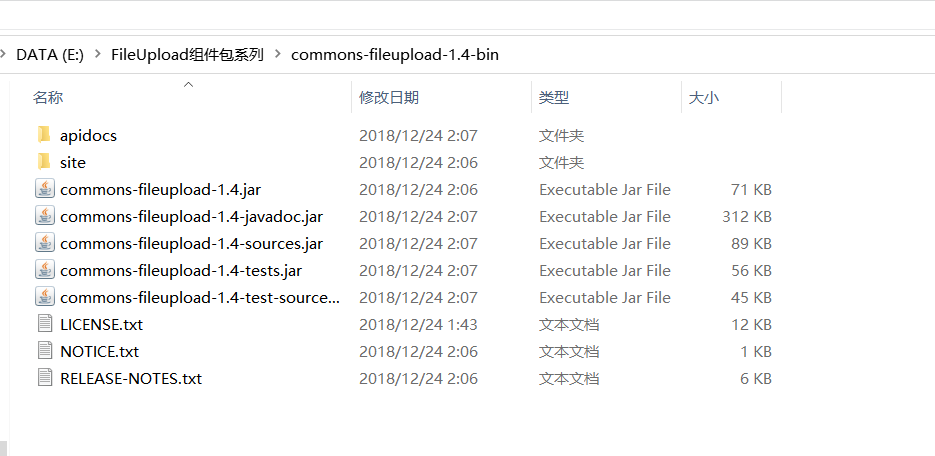
Spring MVC框架的文件上传是基于commons-fileupload组件的文件上传，但Spring MVC框架将上传文件的操作做了进一步的封装，简化了文件上传的代码实现。

## commons-fileupload组件

commons-fileupload组件是Apache的一个子项目，能够实现文件的上传，并支持任意 大小的文件上传。

1. 下载commons-fileupload组件可以从以下地址进行下载：

<http://commons.apache.org/proper/commons-fileupload/>

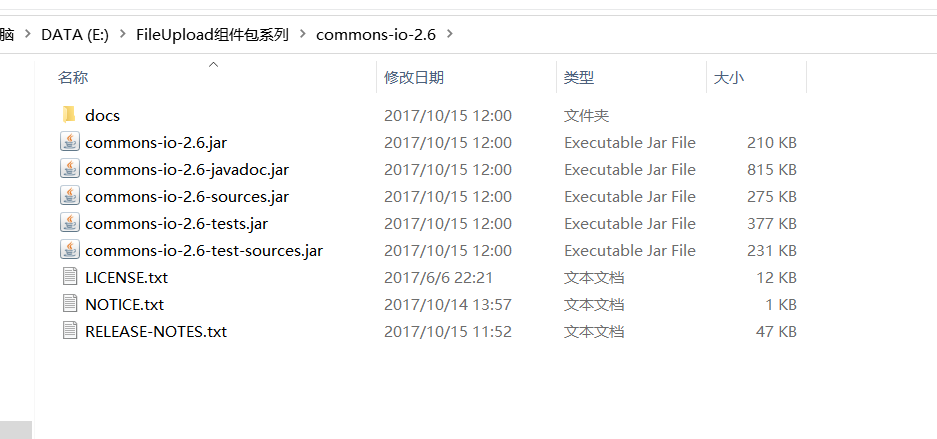


其中最重要的是com.fileupload-1.4.jar，该文件是commons-fileupload组件的类库， 也就是我们要用到的Jar包。

1. 下载commons-io组件

由于commons-fileupload组件依赖于Apache的另外一个项目：commons-io， 所以还需要下载commons-io.jar，该组件可以从以下地址进行下载：

<http://commons.apache.org/proper/commons-io>



其中，最重要的是commons-io-2.6.jar包，该包是我们需要用到的Jar包。

下载完成后，将commons-fileupload1.4.jar包和commons-io-2.6.jar包导入到WEB-INF/lib 目录中。

## 基于表单的文件上传

上传文件的操作是基于表单完成的，使用<input type=”file”/>标签完成，如：

<form action=”” method=”post” entype=”multipart/form-date”>

<input type=”file” name=”myfile”/>

......

</form>

并且通过表单上传文件时，需要使用entype属性，并将属性值设置为multipart/form-data， 同时表单的提交方式必须为post。

## MultipartFile接口

在Spring MVC框架中上传文件时会将文件的相关信息及操作封装到MultipartFile对象中， 因此开发者需要使用MultipartFile来完成文件上传。该接口的常用方法有：

·byte[] getBytes( )：以字节数组的形式返回文件的内容

·String getContentType( )：返回文件的内容类型

·InputStream getInputStream( )：返回一个InputStream，从中读取文件的 内容

·String getName( )：返回请求参数的名称

·String getOrginalFilename( )：返回客户端提交的原始文件名称，包括后缀名

·long getSize( )：返回文件的大小，单位为字节

·boolean isEmpty( )：判断被上传文件是否为空

·void transferTo( File destination )：将上传的文件保存指定目录中。

但要想使用这个接口，还需要在Spring MVC配置文件中进行配置。例如：



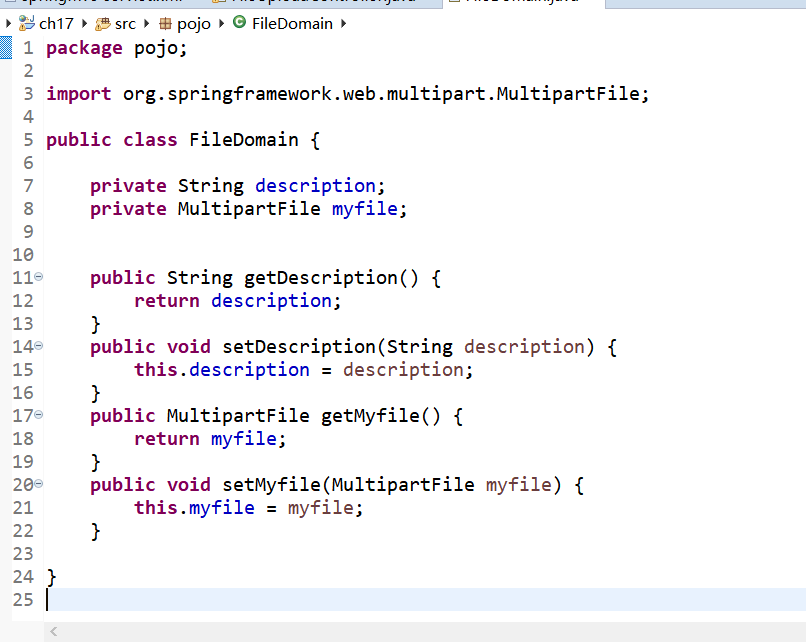
其中，p:defaultEncoding指定请求的编码格式，默认为ISO-8859-8

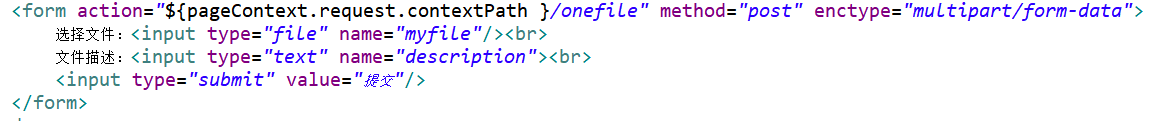
p:maxUploadSize指定上传文件的最大值，单位为字节

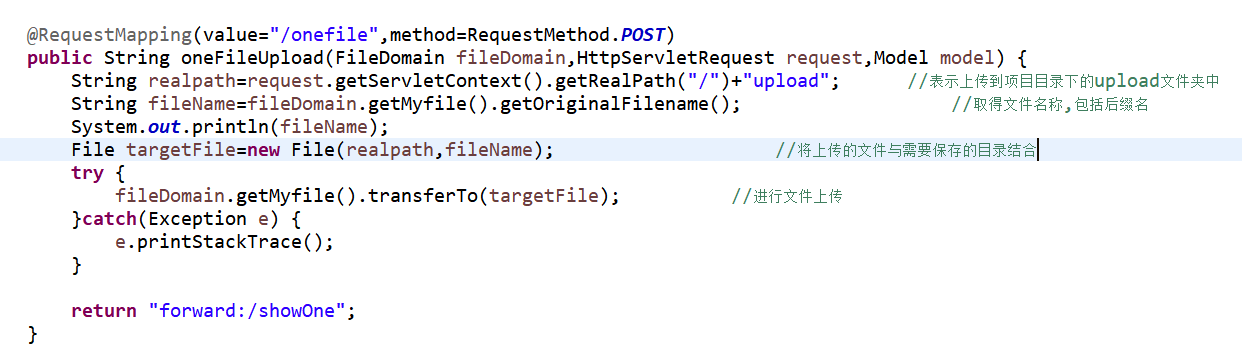
p:uploadTempDir指定上传文件的临时路径，但不是最终保存文件的目录，如不 存在会自动生成

## 上传单个文件

上传单个文件的部分代码如下：



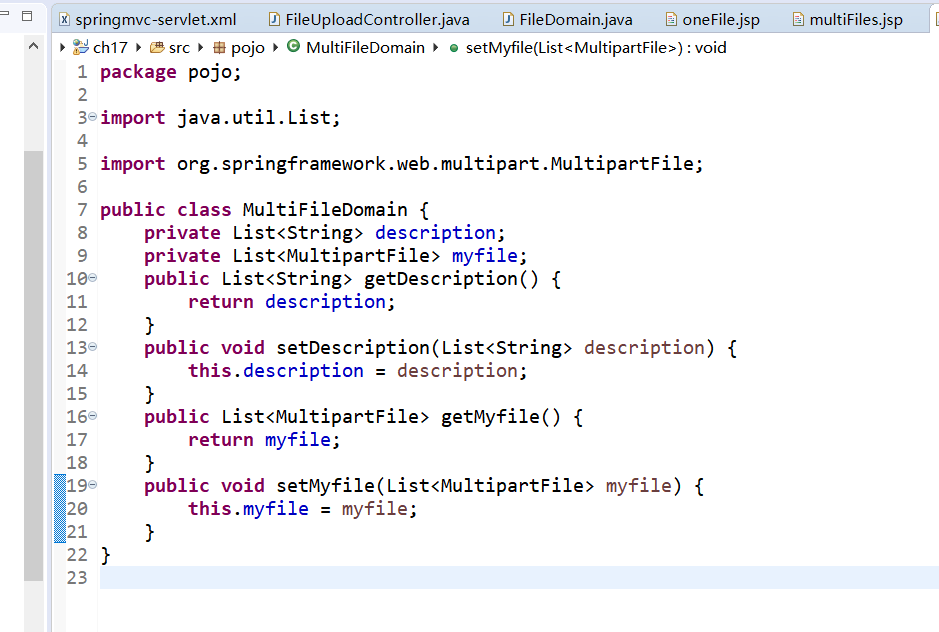


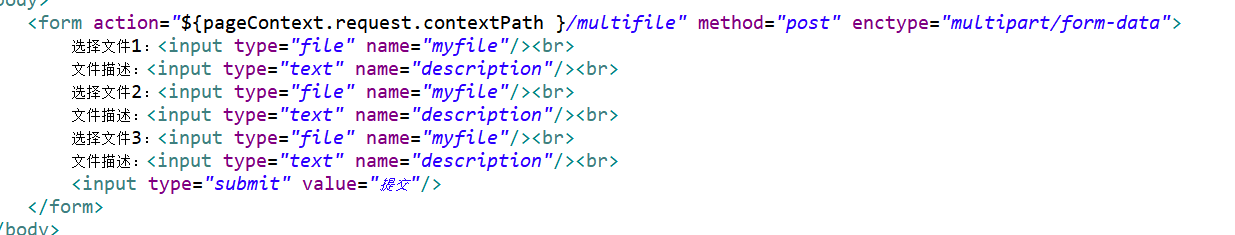




## 上传多个文件

上传多个文件的部分实例代码如下：







配置文件不变



# 文件下载

## 实现思路

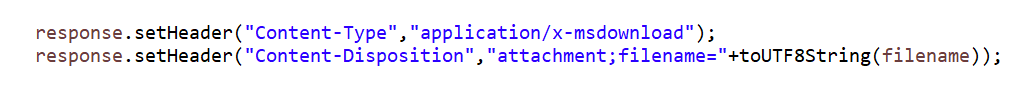
1）得到存放文件目录中的所有文件的文件名，然后将其显示在页面中，并给它们加上 超链接，当点击这些超链接时，就会进入控制器，而控制器调用业务层，在业务层 通过输入输出流完成文件的下载。

2）在进行文件下载的开发时，需要特别注意文件的名称问题，例如在超链接中不支持 传递中文后特殊字符。

## 文件下载的报头要求

在Spring MVC中实现下载需要设置两个报头：

1. Web服务器需要告诉浏览器其所输出内容的类型不是普通文本或HTML文件，而是 一个要保存到本地的下载文件，所以需要设置Content-Type的值为: application/x-msdownload
2. Web服务器不希望浏览器不直接处理相应的实体内容，而是由用户选择将相应的实 体内容保存到一个文件中，这需要设置ContentDisposition报头。该报头指定了接收 程序处理数据内容的方式，在HTTP应用中只有attachment是标准方式，attachment 表示要求用户干预。在attachement后面还可以指定filename参数，该参数时服务 器建议浏览器将实体内容保存到文件中的文件名称。



## 实现文件下载

实现文件下载的核心代码如下：

