



链滴

# Spring Boot 文件上传

作者: [mtkx](#)

原文链接: <https://ld246.com/article/1555452880098>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

## 1. 首先创建一个Spring Boot项目并添spring-boot-starter-web 赖。

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>
```

## 2. 然后在resources录下的static目录中创建一个upload.html 文件 内容如下:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<form action="/upload" method="post" enctype="multipart/form-data">
  <input type="file" name="uploadFile" value="请选择文件">
  <input type="submit" value="上传">
</form>
</body>
</html>
```

这是一个很简单的文件上传页面，上传接口是/upload注意请求方法是post，enctype是multipart/form-data

## 3. 接着创建文件上传处理接口，代码如下:

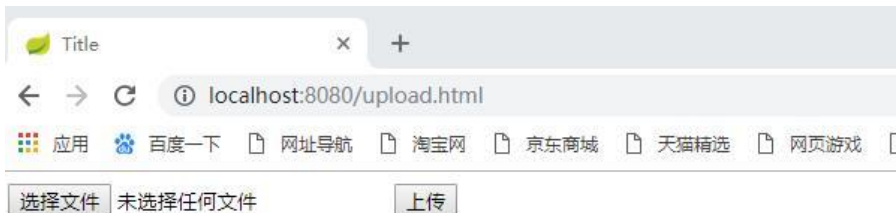
```
@RestController
```

```
public class FileUploadController {
    SimpleDateFormat sdf=new SimpleDateFormat("yyyy/MM/dd");

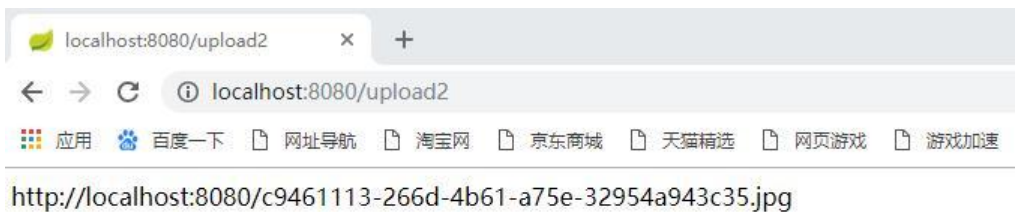
    @PostMapping("/upload")
    public String upload2(MultipartFile uploadFile, HttpServletRequest req){
        //获取文件名
        String fileName = uploadFile.getOriginalFilename();
        //获取文件后缀名
        String suffixName = fileName.substring(fileName.lastIndexOf("."));
        //重新生成文件名
        fileName = UUID.randomUUID()+suffixName;
        //指定本地文件夹存储图片
        String filePath = "E:\\eclipse-workspace\\chapter\\bv_4_1\\src\\main\\resources\\static\\";
;
        try {
            //将图片保存到static文件夹里
            uploadFile.transferTo(new File(filePath+fileName));
            String filePath2=req.getScheme()+"://"+req.getServerName()+":"+req.getServerPort()+"/"+fileName;
            //返回图片地址
```

```
        return filePath2;
    } catch (Exception e) {
        e.printStackTrace();
        return "失败";
    }
}
}
```

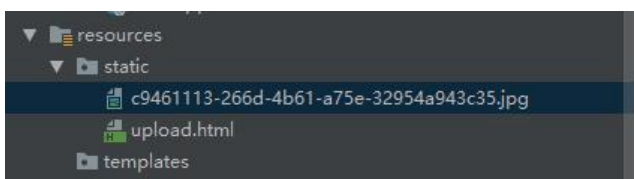
## 4. 运行项目，在浏览器中输入 `http://localhost:8080/upload.html` 进行文件上传



### 上传成功界面:



### 刷新static出现对应的文件:



## 5. 大文件上传失败（解决办法）

以上上传文件限制为1M以内,我们可以通过配置文件进行设置。源码如下:

```
#是否开启文件上传支持，默认为 true
#spring.servlet.multipart.enabled=true
#文件写入磁盘的阈值，默认为0
#spring.servlet.multipart.file-size-threshold=0
#上传文件的临时保存位置
#spring.servlet.multipart.location=E:\\temp
#是否延迟解析
#spring.servlet.multipart.resolve-lazily=false
```

```
#上传文件最大
spring.servlet.multipart.max-file-size=10MB
#多文件上传最大总值
spring.servlet.multipart.max-request-size=100MB
```

至此，一个简单的图片上传逻辑就完成了，对于开发者而，只需要专注于图片上传的业务逻辑，而需要在配置上花费太多时间。