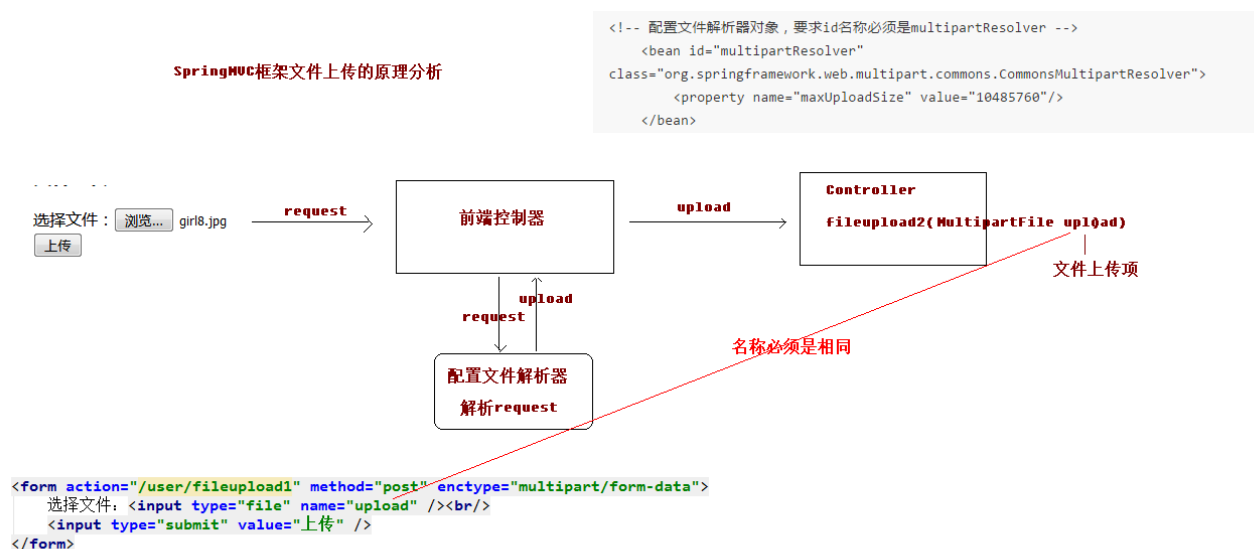# springmvc文件上传

## 一、基本介绍

### 1、主要知识点

主要通过文件解析器对象org.springframework.web.multipart.commons.CommonsMultipartResolver来实现了传统文件的解析功能等等。

可以去看它的源码

### 2、原理图解



### 3、注意点

#### 1）文件解析器的id必须固定multipartResolver

```xml
<!--配置文件解析器对象-->
  <bean id="multipartResolver" class="org.springframework.web.multipart.commons.CommonsMultipartResolver">
    <property name="maxUploadSize" value="10485760" />
  </bean>
```

#### 2）文件表单的属性名必须和MultipartFile方法形参名一样

```html
<input type="file" name="upload" /><br/>
```

```java
public String fileuoload2(HttpServletRequest request, MultipartFile upload) throws Exception
```

## 二、实现步骤

### 1、配置文件解析器对象

```xml
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
    xmlns:mvc="http://www.springframework.org/schema/mvc"
    xmlns:context="http://www.springframework.org/schema/context"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="
     http://www.springframework.org/schema/beans
     http://www.springframework.org/schema/beans/spring-beans.xsd
     http://www.springframework.org/schema/mvc
     http://www.springframework.org/schema/mvc/spring-mvc.xsd
     http://www.springframework.org/schema/context
     http://www.springframework.org/schema/context/spring-context.xsd">

  <!-- 开启注解扫描 -->
  <context:component-scan base-package="com.qiu"/>

  <!-- 视图解析器对象 -->
```

```xml
    <bean id="internalResourceViewResolver"
class="org.springframework.web.servlet.view.InternalResourceViewResolver">
        <property name="prefix" value="/WEB-INF/pages/"/>
        <property name="suffix" value=".jsp"/>
    </bean>

    <!--前端控制器，哪些静态资源不拦截-->
    <mvc:resources location="/js/" mapping="/js/**"/>

    <!--配置文件解析器对象-->
    <bean id="multipartResolver" class="org.springframework.web.multipart.commons.CommonsMultipartResolver">
        <property name="maxUploadSize" value="10485760" />
    </bean>

    <!-- 开启SpringMVC框架注解的支持 -->
    <mvc:annotation-driven />

</beans>
```

## 2、配置表单操作页面

```jsp
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<html>
<head>
    <title>Title</title>

</head>
<body>
  <h3>Springmvc文件上传</h3>

    <form action="user/fileupload2" 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、配置文件上传解析

```java
@Controller
@RequestMapping("/user")
public class UserController {

    /**
     * 文件上传
     * @return
     */
    @RequestMapping("/fileupload2")
    public String fileuoload2(HttpServletRequest request, MultipartFile upload) throws Exception {
        System.out.println("springmvc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();
        }

        // 说明上传文件项
        // 获取上传文件的名称
        String filename = upload.getOriginalFilename();
        // 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
```

```
        upload.transferTo(new File(path,filename));

        return "success";
    }
}
```