# SpringMVC-02

### 学习目标

* + - 掌握SpringMVC的响应数据和结果视图
    - 掌握SpringMVC实现文件上传
    - 掌握SpringMVC的异常处理
    - 了解SpringMVC中的拦截器

### 课程内容

* SpringMVC响应数据和结果视图
* SpringMVC实现文件上传
* SpringMVC的异常处理
* SpringMVC框架中的拦截器

### 第一堂课

本节知识点

1.1 搭建环境

1.2 响应之返回值是String类型

1.3 响应之返回值是void类型

1.4 响应之返回值是ModelAndView类型

视频时间

29分14秒

本节目标

掌握SpringMVC响应返回值类型

#### 1.1 搭建环境

##### 1.1.1知识概述

本节课主要讲解今天课程内容的环境搭建。

##### 1.1.2视频详情



##### 1.1.3总结与补充

开发环境搭建

##### 1.1.4课堂提问与练习

无

##### 1.1.5习题答案

无

##### 1.1.6练习答案

无

##### 1.1.7视频缺陷

无

##### 1.1.8视频扩展

无

#### 1.2 响应之返回值是String类型

##### 1.2.1知识概述

本节课主要讲解演示SpringMVC中方法返回值是String类型。

##### 1.2.2视频详情



##### 1.2.3总结与补充

Controller方法返回字符串可以指定逻辑视图的名称，根据视图解析器为物理视图的地址。

##### 1.2.4课堂提问与练习

无

##### 1.2.5习题答案

无

##### 1.2.6练习答案

无

##### 1.2.7视频缺陷

无

##### 1.2.8视频扩展

无

#### 1.3 响应之返回值是void类型

##### 1.3.1知识概述

本节课主要讲解演示SpringMVC中方法返回值是void类型。

##### 1.3.2视频详情



##### 1.3.3总结与补充

返回值是void

1. 如果控制器的方法返回值编写成void，执行程序报404的异常，默认查找JSP页面没有找到。默认会跳转到@RequestMapping(value="/initUpdate") initUpdate的页面。

2. 可以使用请求转发或者重定向跳转到指定的页面

##### 1.3.4课堂提问与练习

无

##### 1.3.5习题答案

无

##### 1.3.6 练习答案

无

##### 1.3.7 视频缺陷

无

##### 1.3.8 视频扩展

无

#### 1.4 响应之返回值是ModelAndView类型

##### 1.4.1知识概述

本节课主要讲解演示SpringMVC中方法返回值是ModelAndView类型。

##### 1.4.2视频详情



##### 1.4.3总结与补充

返回值是ModelAndView对象

ModelAndView对象是Spring提供的一个对象，可以用来调整具体的JSP视图

##### 1.4.4课堂提问与练习

无

##### 1.4.5习题答案

无

##### 1.4.6 练习答案

无

##### 1.4.7 视频缺陷

无

##### 1.4.8 视频扩展

无

### 第二堂课

本节知识点：

2.1 响应之使用forward和redirect进行页面跳转

2.2 响应json数据之过滤静态资源

2.3 响应json数据之发送ajax的请求

2.4 响应json数据之响应json格式数据

视频时间：

31分03秒

本节目标 ：

掌握SpringMVC如何实现页面跳转

掌握SpringMVC如何响应json数据

#### 2.1 响应之使用forward和redirect进行页面跳转

##### 2.1.1知识概述

本节课主要讲解SpringMVC如何实现页面跳转。

##### 2.1.2视频详情



##### 2.1.3总结与补充

1. forward请求转发

controller方法返回String类型，想进行请求转发也可以编写成: return” forward:/user/findAll”

2. redirect重定向

controller方法返回String类型，想进行请求转发也可以编写成:return” redirect:/user/findAll”

##### 2.1.4课堂提问与练习

无

##### 2.1.5习题答案

无

##### 2.1.6 练习答案

无

##### 2.1.7 视频缺陷

无

##### 2.1.8 视频扩展

无

#### 2.2 响应json数据之过滤静态资源

##### 2.2.1知识概述

本节课主要讲解SrpingMVC中如何配置静态资源映射，也就是不拦截静态资源。

##### 2.2.2视频详情



##### 2.2.3总结与补充

DispatcherServlet会拦截到所有的资源，导致一个问题就是静态资源（img、css、js）也会被拦截到，从而

不能被使用。解决问题就是需要配置静态资源不进行拦截，在springmvc.xml配置文件添加如下配置

1. mvc:resources标签配置不过滤

1. location元素表示webapp目录下的包下的所有文件

2. mapping元素表示以/static开头的所有请求路径，如/static/a 或者/static/a/b

2. 使用@RequestBody获取请求体数据

@RequestMapping("/count")

public String count() throws Exception {

System.out.println("count方法执行了...");

return "redirect:/add.jsp";

// return "redirect:/user/findAll";

}

<!-- 设置静态资源不过滤 -->

<mvc:resources location="/css/" mapping="/css/\*\*"/> <!-- 样式 -->

<mvc:resources location="/images/" mapping="/images/\*\*"/> <!-- 图片 -->

<mvc:resources location="/js/" mapping="/js/\*\*"/> <!-- javascript -->

##### 2.2.4课堂提问与练习

无

##### 2.2.5习题答案

无

##### 2.2.6 练习答案

无

##### 2.2.7 视频缺陷

无

##### 2.2.8 视频扩展

无

#### 2.4 响应json数据之发送ajax的请求

##### 2.4.1知识概述

本节主要讲解使用ajax发送给一个异步请求，并且将请求参数封装到javabean中。

##### 2.4.2视频详情



##### 2.4.3总结与补充

无

##### 2.4.4课堂提问与练习

无

##### 2.4.5习题答案

无

##### 2.4.6 练习答案

无

##### 2.4.7 视频缺陷

无

##### 2.4.8 视频扩展

无

### 第三堂课

本节知识点：

3.1 文件上传之上传原理分析和搭建环境

3.2 文件上传之传统方式上传代码回顾

3.3 文件上传之Springmvc方式上传原理分析

视频时间：

33分45秒

本节目标 ：

回顾传统文件上传实现方式

掌握SpringMVC文件上传原理分析

#### 3.1文件上传之上传原理分析和搭建环境

##### 3.1.1知识概述

本节课主要回顾传统文件上传方式的上传和开发环境搭建。

##### 3.1.2视频详情



##### 3.1.3总结与补充

传统文件上传需要注意事项：

1、设置enctype

2、post请求方式

3、文件上传表单项

##### 3.1.4课堂提问与练习

无

##### 3.1.5习题答案

无

##### 3.1.6练习答案

无

##### 3.1.7视频缺陷

无

##### 3.1.8视频扩展

无

#### 3.2 文件上传之传统方式上传代码回顾

##### 3.2.1知识概述

回顾传统文件上传代码。

##### 3.2.2视频详情



##### 3.2.3总结与补充

无

##### 3.2.4课堂提问与练习

无

##### 3.2.5习题答案

无

##### 练习答案

无

##### 3.2.7视频缺陷

无

##### 3.2.8视频扩展

无

#### 3.3 文件上传之SpringMVC方式上传原理分析

##### 3.3.1知识概述

本节课主要讲解SpringMVC如何实现文件上传的原理分析。

##### 3.3.2视频详情



##### 3.3.3总结与补充

SpringMVC框架提供了MultipartFile对象，该对象表示上传的文件，要求变量名称必须和表单file标签的name属性名称相同。

##### 3.3.4课堂提问与练习

无

##### 3.3.5习题答案

无

##### 3.3.6 练习答案

无

##### 3.3.7 视频缺陷

无

##### 3.3.8 视频扩展

无

### 第四堂课

本节知识点：

4.1 文件上传之SpringMVC方式上传代码

4.2 文件上传之跨服务器上传分析和搭建环境

4.3 文件上传之跨服务器上传代码

视频时间

23分06秒

本节目标 ：

掌握SpringMVC实现跨服务器文件上传

#### 4.1 4.1 文件上传之SpringMVC方式上传代码

##### 4.1.1知识概述

本节课主要是SpringMVC方式上传代码的编写。

##### 4.1.2视频详情



##### 4.1.3总结与补充

无

##### 4.1.4课堂提问与练习

无

##### 4.1.5习题答案

无

##### 练习答案

无

##### 视频缺陷

无

##### 视频扩展

无

#### 4.2 文件上传之跨服务器上传分析和搭建环境

##### 4.2.1知识概述

本节课主要讲解跨服务器文件上传的分析和环境搭建。

##### 4.2.2视频详情



##### 4.2.3总结与补充

无

##### 4.2.4课堂提问与练习

无

##### 4.2.5习题答案

无

##### 练习答案

无

##### 4.2.7视频缺陷

注意：后面老师演示的时候服务器没有这个上传的uploads路径。老师手动创建的目录。这个问题的解决方案可以通过自己写一个监听器来实现。实现方案比较简单这里就不在贴出代码。各位老师自己可以自由发挥。

##### 4.2.8视频扩展

无

#### 4.3 文件上传之跨服务器上传代码

##### 4.3.1知识概述

本节课主要讲解文件上传跨服务器代码实现。

##### 4.3.2视频详情



##### 4.3.3总结与补充

无

##### 4.3.4课堂提问与练习

无

##### 4.3.5习题答案

无

##### 4.3.6练习答案

无

##### 4.3.7视频缺陷

无

##### 4.3.8视频扩展

无

### 第五堂课

本节知识点：

5.1 SpringMVC异常处理之分析和搭建环境

5.2 SpringMVC异常处理之演示程序异常

5.3 SpringMVC异常处理之异常处理代码编写

视频时间

27分44秒

本节目标 ：

掌握SpringMVC异常处理。

#### 5.1 SpringMVC异常处理之分析和搭建环境

##### 5.1.1知识概述

. 本节课主要讲解SpringMVC异常处理的思路和开发环境的搭建

##### 5.1.2视频详情



##### 5.1.3总结与补充

无

##### 5.1.4课堂提问与练习

无

##### 5.1.5习题答案

无

##### 5.1.6练习答案

无

##### 5.1.7视频缺陷

无

##### 5.1.8视频扩展

无

#### 5.2 SpringMVC异常处理之演示程序异常

##### 5.2.1知识概述

本节课主要演示程序异常。

##### 5.2.2视频详情



##### 5.2.3总结与补充

无

##### 5.2.4课堂提问与练习

无

##### 5.2.5习题答案

无

##### 5.2.6练习答案

无

##### 5.2.7视频缺陷

无

##### 5.2.8视频扩展

无

#### 5.3 SpringMVC异常处理之异常处理代码编写

##### 5.3.1知识概述

本节课主要演示SpringMVC如何处理异常代码。

##### 5.3.2视频详情



##### 5.3.3总结与补充

无

##### 5.3.4课堂提问与练习

无

##### 5.3.5习题答案

无

##### 5.3.6练习答案

无

##### 5.3.7视频缺陷

无

##### 5.3.8视频扩展

无

### 第六堂课

本节知识点：

6.1 SpringMVC拦截器之介绍和搭建环境

6.2 SpringMVC拦截器之编写controller

6.3 SpringMVC拦截器之拦截器入门代码

6.4 SpringMVC拦截器之拦截器接口方法演示

视频时间

31分49秒

本节目标 ：

熟练掌握SpringMVC拦截器的配置和使用

#### 6.1 SpringMVC拦截器之介绍和搭建环境

##### 6.1.1知识概述

1. SpringMVC框架中的拦截器用于对处理器进行预处理和后处理的技术。

2. 可以定义拦截器链，连接器链就是将拦截器按着一定的顺序结成一条链，在访问被拦截的方法时，拦截器链

中的拦截器会按着定义的顺序执行。

3. 拦截器和过滤器的功能比较类似，有区别

1. 过滤器是Servlet规范的一部分，任何框架都可以使用过滤器技术。

2. 拦截器是SpringMVC框架独有的。

3. 过滤器配置了/\*，可以拦截任何资源。

4. 拦截器只会对控制器中的方法进行拦截。

4. 拦截器也是AOP思想的一种实现方式

5. 想要自定义拦截器，需要实现HandlerInterceptor接口。

##### 6.1.2视频详情



##### 6.1.3总结与补充

1. SpringMVC框架中的拦截器用于对处理器进行预处理和后处理的技术。

2. 可以定义拦截器链，连接器链就是将拦截器按着一定的顺序结成一条链，在访问被拦截的方法时，拦截器链

中的拦截器会按着定义的顺序执行。

3. 拦截器和过滤器的功能比较类似，有区别

1. 过滤器是Servlet规范的一部分，任何框架都可以使用过滤器技术。

2. 拦截器是SpringMVC框架独有的。

3. 过滤器配置了/\*，可以拦截任何资源。

4. 拦截器只会对控制器中的方法进行拦截。

4. 拦截器也是AOP思想的一种实现方式

5. 想要自定义拦截器，需要实现HandlerInterceptor接口。

##### 6.1.4课堂提问与练习

无

##### 6.1.5习题答案

无

##### 6.1.6练习答案

无

##### 6.1.7视频缺陷

无

##### 6.1.8视频扩展

无

#### 6.2 SpringMVC拦截器之编写Controller

##### 6.2.1知识概述

本节课主要演示 SpringMVC拦截器之编写Controller

##### 6.2.2视频详情



##### 6.2.3总结与补充

无

##### 6.2.4课堂提问与练习

无

##### 6.2..5习题答案

无

##### 6.2.6练习答案

无

##### 6.2.7视频缺陷

无

##### 6.2.8视频扩展

无

#### 6.3 SpringMVC拦截器之拦截器入门代码

##### 6.3.1知识概述

本节课主要讲解拦截器入门代码

##### 6.3.2视频详情



##### 6.3.3总结与补充

无

##### 6.3.4课堂提问与练习

无

##### 6.3..5习题答案

无

##### 6.3.6练习答案

无

##### 6.3.7视频缺陷

无

##### 6.3.8 视频扩展

无

#### 6.4 SpringMVC拦截器之拦截器接口方法演示

##### 6.4.1知识概述

主要讲解拦截器效果，如何配置多个拦截器？

##### 6.4.2视频详情



##### 6.4.3总结与补充

##### 6.4.4课堂提问与练习

无

##### 6.4..5习题答案

无

##### 6.4.6练习答案

无

##### 6.4.7视频缺陷

无

##### 6.4.8 视频扩展

无