# springmvc-01

### 学习目标

1. 了解SpringMVC的基本概念
2. 掌握SpringMVC的入门程序
3. 掌握SpringMVC中请求参数的绑定
4. 掌握SpringMVC中其他常用注解

### 课程内容

* 1. SpringMVC框架的介绍入门程序
* 2. SpringMVC请求参数绑定
* 3. SpringMVC其他常用注解

### 第一堂课

本节知识点

1.1 三层架构介绍和MVC设计模型介绍

1.2 SpringMVC框架的介绍

视频时间

21分35秒

本节目标

了解EE的三层架构和MVC设计模型

了解SpringMVC框架基本介绍

#### 1.1 三层架构介绍和MVC设计模型介绍

##### 1.1.1知识概述

本节课主要讲解回顾EE的三层结构和MVC设计模型，主要以复习回顾引导为主。

##### 1.1.2视频详情



##### 1.1.3总结与补充

EE的三层结构：

* 表现层【WEB层】
* 业务层【SERVICE层】
* 持久层【DAO层】

MVC设计模型：

* 控制器【C】
* 模型【M】
* 视图【V】

##### 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 SpringMVC框架的介绍

##### 1.2.1知识概述

本节课主要讲解SpringMVC在EE三层结构中的位置，以及SpringMVC能够处理的问题，它是一个经典的MVC框架。它与spring无缝衔接，主流的互联网企业都在使用它。本节课涉及三个问题：1.SpringMVC是什么?2.SpringMVC框架所处的位置？3.SpringMVC的优势？

##### 1.2.2视频详情



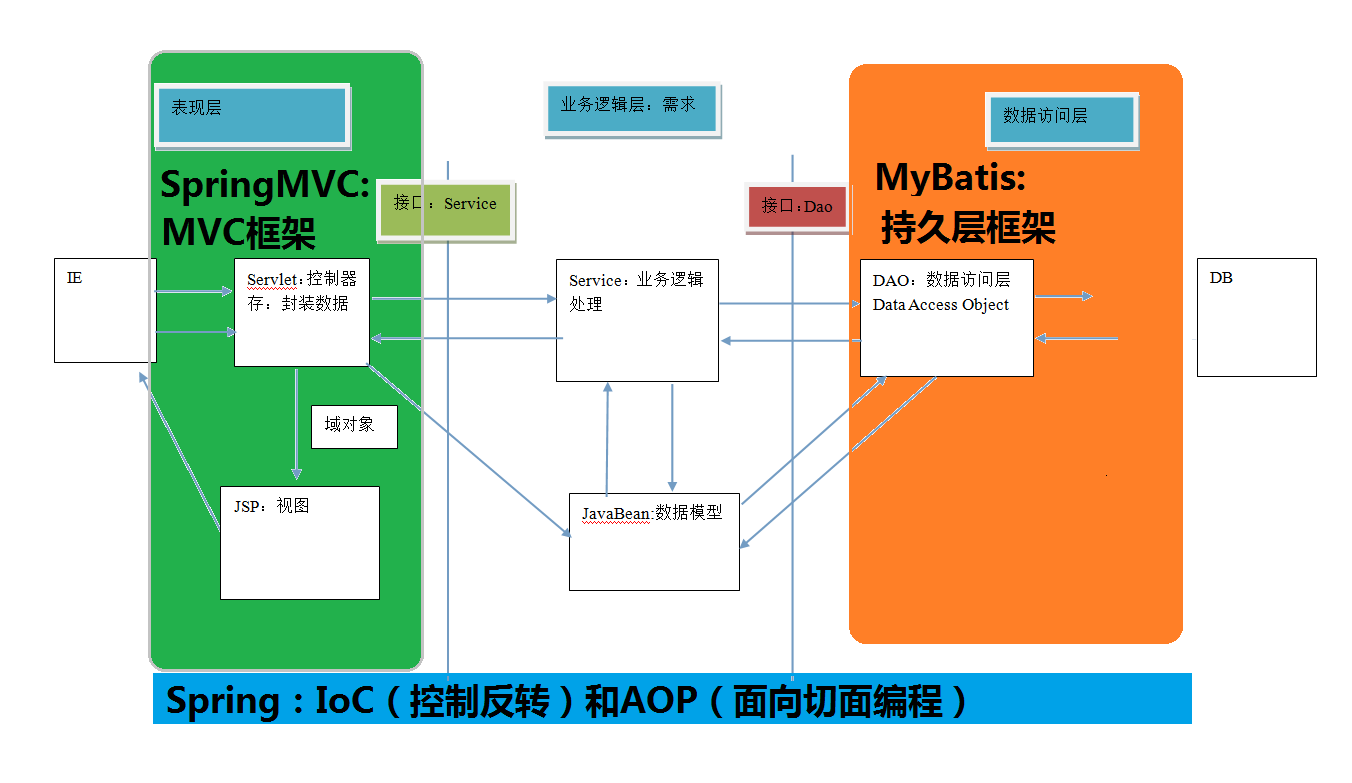
##### 1.2.3总结与补充

1、SpringMVC是什么？

SpringMVC 是一种基于 Java 的实现 MVC 设计模型的请求驱动类型的轻量级 Web 框架，属于 SpringFrameWork 的后续产品，已经融合在 Spring Web Flow 里面。Spring 框架提供了构建 Web 应用程序的全功能 MVC 模块。使用 Spring 可插入的 MVC 架构，从而在使用 Spring 进行 WEB 开发时，可以选择使用 Spring的 Spring MVC 框架或集成其他 MVC 开发框架，如 Struts1(现在一般不用)，Struts2 等。SpringMVC 已经成为目前最主流的 MVC 框架之一，并且随着 Spring3.0 的发布，全面超越 Struts2，成为最优秀的 MVC 框架。它通过一套注解，让一个简单的 Java 类成为处理请求的控制器，而无须实现任何接口。同时它还支持RESTful 编程风格的请求。

2、SpringMVC所处的位置？

SpringMVC处于EE三层结中的WEB层。



3、SpringMVC的优势

1、清晰的角色划分：

前端控制器（DispatcherServlet）

请求到处理器映射（HandlerMapping）

处理器适配器（HandlerAdapter）

视图解析器（ViewResolver）

处理器或页面控制器（Controller）

验证器（ Validator）

命令对象（Command 请求参数绑定到的对象就叫命令对象）

表单对象（Form Object 提供给表单展示和提交到的对象就叫表单对象）。

2、分工明确，而且扩展点相当灵活，可以很容易扩展，虽然几乎不需要。

3、由于命令对象就是一个 POJO，无需继承框架特定 API，可以使用命令对象直接作为业务对象。

4、和 Spring 其他框架无缝集成，是其它 Web 框架所不具备的。

5、可适配，通过 HandlerAdapter 可以支持任意的类作为处理器。

6、可定制性，HandlerMapping、ViewResolver 等能够非常简单的定制。

7、功能强大的数据验证、格式化、绑定机制。

8、利用 Spring 提供的 Mock 对象能够非常简单的进行 Web 层单元测试。

9、本地化、主题的解析的支持，使我们更容易进行国际化和主题的切换。

10、强大的 JSP 标签库，使 JSP 编写更容易。

………………还有比如RESTful风格的支持、简单的文件上传、约定大于配置的契约式编程支持、基于注解的零配

置支持等等。

##### 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

### 第二堂课

本节知识点：

2.1 入门程序之需求分析

2.2 入门程序之搭建开发环境

2.3 入门程序之入门代码编写

2.4 入门案例的流程总结

视频时间：

38分10秒

本节目标 ：

了解SpringMVC入门程序的需求

掌握SpringMVC开发环境的搭建和入门代码编写

理解SpringMVC的执行流程

#### 2.1 入门程序之需求分析

##### 2.1.1知识概述

本节课主要讲解SpringMVC入门案例的 需求：发送一个请求给后台程序，后台能够接收到该请求并且执行对应的业务方法，输出一段话。并且返回一个成功的页面。

##### 2.1.2视频详情



##### 2.1.3总结与补充

无

##### 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 入门程序之搭建开发环境

##### 2.2.1知识概述

本节课主要讲解SpringMVC入门程序开发环境的搭建。

##### 2.2.2视频详情



##### 2.2.3总结与补充

开发环境搭建：

1. 复制相关依赖
2. web.xml中配置SpringMVC的前端控制器
3. 配置SpringMVC的配置文件：springmvc.xml
4. 编写Controller代码
5. 编写成功页面
6. 测试

##### 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

#### 2.3 入门程序之入门代码编写

##### 2.3.1知识概述

本节课主要讲解SpringMVC中的核心组件处理器的代码编写。

##### 2.3.2视频详情



##### 2.3.3总结与补充

编写一个HelloController

/\*\*

\* 控制器

\*/

@Controller

public class HelloController {

/\*\*

\* 接收请求

\* @return

\*/

@RequestMapping(path="/hello")

public String sayHello() {

System.out.println("Hello SpringMVC!!");

return "success";

}

}

##### 2.3.4课堂提问与练习

无

##### 2.3.5习题答案

无

##### 2.3.6 练习答案

无

##### 2.3.7 视频缺陷

无

##### 2.3.8 视频扩展

无

#### 2.4 入门案例的流程总结

##### 2.4.1知识概述

本节课主要是对入门案例的总结，分析SpringMVC的执行流程。

##### 2.4.2视频详情



##### 2.4.3总结与补充

入门案例的执行流程：

1. 当启动Tomcat服务器的时候，因为配置了load-on-startup标签，所以会创建DispatcherServlet对象，

就会加载springmvc.xml配置文件

2. 开启了注解扫描，那么HelloController对象就会被创建

3. 从index.jsp发送请求，请求会先到达DispatcherServlet核心控制器，根据配置@RequestMapping注解

找到执行的具体方法

4. 根据执行方法的返回值，再根据配置的视图解析器，去指定的目录下查找指定名称的JSP文件

5. Tomcat服务器渲染页面，做出响应

##### 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 入门案例中使用的组件介绍

3.2 RequestMapping注解的作用

3.3 RequestMapping注解的属性

视频时间：

28分45秒

本节目标 ：

了解SpringMVC内部组件

掌握@RequestMapping注解的属性及其作用和使用

#### 3.1 入门案例中使用的组件介绍

##### 3.1.1知识概述

本节课主要讲解入门案例中SpringMVC的各个组件及其作用。

##### 3.1.2视频详情



##### 3.1.3总结与补充

1. 前端控制器（DispatcherServlet）

2. 处理器映射器（HandlerMapping）

3. 处理器（Handler）

4. 处理器适配器（HandlAdapter）

5. 视图解析器（View Resolver）

6. 视图（View）

##### 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 RequestMapping注解的作用

##### 3.2.1知识概述

本节课主要讲解@RequestMapping 注解的作用以及使用方法。

##### 3.2.2视频详情



##### 3.2.3总结与补充

1. RequestMapping注解的作用是建立请求URL和处理方法之间的对应关系

2. RequestMapping注解可以作用在方法和类上

1. 作用在类上：第一级的访问目录
2. 作用在方法上：第二级的访问目录
3. 细节：路径可以不编写 / 表示应用的根目录开始
4. 细节：${ pageContext.request.contextPath }也可以省略不写，但是路径上不能写 /

##### 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 RequestMapping注解的属性

##### 3.3.1知识概述

本节课主要讲解SpringMVC中@RequestMapping注解的属性及其用法。

##### 3.3.2视频详情



##### 3.3.3总结与补充

RequestMapping的属性

1. path 指定请求路径的url

2. value value属性和path属性是一样的

3. mthod 指定该方法的请求方式

4. params 指定限制请求参数的条件

5. headers 发送的请求中必须包含的请求头

##### 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 请求参数绑定入门

4.2 请求参数绑定实体类型

4.3 配置解决中文乱码的过滤器

4.4 请求参数绑定集合类型

视频时间

29分06秒

本节目标 ：

掌握使用SpringMVC的参数绑定

#### 4.1 请求参数绑定入门

##### 4.1.1知识概述

本节课主要讲解SpringMVC的参数绑定，也就是如何把前台传递过来的参数接收到并且封装起来给程序使用。

##### 4.1.2视频详情



##### 4.1.3总结与补充

1. 绑定机制

1. 表单提交的数据都是k=v格式的 username=haha&password=123

2. SpringMVC的参数绑定过程是把表单提交的请求参数，作为控制器中方法的参数进行绑定的

3. 要求：提交表单的name和参数的名称是相同的

2. 支持的数据类型

1. 基本数据类型和字符串类型

2. 实体类型（JavaBean）

3. 集合数据类型（List、map集合等）4.1.4课堂提问与练习

##### 4.1.5习题答案

无

##### 4.1.6练习答案

无

##### 4.1.7视频缺陷

无

##### 4.1.8视频扩展

无

#### 4.2 请求参数绑定实体类型

##### 4.2.1知识概述

本节课主要讲解，如果前端传递多个参数，这个时候一般会做一个实体来接收这个参数，把请求参数封装到实体中。

##### 4.2.2视频详情



##### 4.2.3总结与补充

1. 提交表单的name和JavaBean中的属性名称需要一致

2. 如果一个JavaBean类中包含其他的引用类型，那么表单的name属性需要编写成：对象.属性 例如：address.name

##### 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

无

##### 4.2.8视频扩展

无

#### 4.3配置解决中文乱码的过滤器

##### 4.3.1知识概述

本节课主要讲解，如果请求参数中携带了中文会产生中文乱码问题，这个时候我们配置类乱码过滤器就可以解决POST方式的中文乱码问题，但是get请求需要手动处理。【tomct8以上版本get请求解决了中文乱码问题】

##### 4.3.2视频详情



##### 4.3.3总结与补充

|  |
| --- |
| !-- 配置过滤器，解决中文乱码的问题 -->  <filter>  <filter-name>characterEncodingFilter</filter-name>  <filter-class>org.springframework.web.filter.CharacterEncodingFilter</filter-  class>  <!-- 指定字符集 -->  <init-param>  <param-name>encoding</param-name>  <param-value>UTF-8</param-value>  </init-param>  </filter>  <filter-mapping>  <filter-name>characterEncodingFilter</filter-name>  <url-pattern>/\*</url-pattern>  </filter-mapping> |

##### 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

#### 4.4 配置解决中文乱码的过滤器

##### 4.4.1知识概述

本节课主要讲解如何将参数封装到集合中。

##### 4.4.2视频详情



##### 4.4.3总结与补充

##### 4.4.4课堂提问与练习

无

##### 4.4.5习题答案

无

##### 4.4.6练习答案

无

##### 4.4.7视频缺陷

无

##### 4.4.8视频扩展

无

### 第五堂课

本节知识点：

5.1 自定义类型转换器演示异常

5.2 自定义类型转换器代码编写

5.3 获取Servlet原生的API

视频时间

23分29秒

本节目标 ：

了解自定义类型转换器

掌握如何获取Servelt原生API

#### 5.1自定义类型转换器演示异常

##### 5.1.1知识概述

在springMVC封装传递的参数有时候需要自己写一个类型转换器来实现，本节课就是讲解如何自定义类型转换器。

##### 5.1.2视频详情



##### 5.1.3总结与补充

自定义类型转换器步骤：

1. 实现Converter接口
2. SpringMVC的配置文件中注册转换器

##### 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 自定义类型转换器代码编写

##### 5.2.1知识概述

本节课主要讲解自定以类型转换器的代码编写。

##### 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 获取Servlet原生的API

##### 5.3.1知识概述

SpringMVC是一个WEB层框架，那么在使用过程中难以避免的使用Servelt的相关API怎么办？只需要在控制器的方法参数定义HttpServletRequest和HttpServletResponse对象

##### 5.3.2视频详情



##### 5.3.3总结与补充

只需要在控制器的方法参数定义HttpServletRequest和HttpServletResponse对象

##### 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 @RequestParam注解

6.2 @RequestBody注解

6.3 @PathVariable注解

6.4 @HiddentHttpMethodFilter过滤器

6.5 @RequestHeader注解

6.6 @CookieValue注解

6.7 @ModelAttribute注解

6.8 @SessionAttributes注解

视频时间

53分19秒

本节目标 ：

熟练掌握SpringMVC中其他常用注解

#### 6.1 @RequestParam注解

##### 6.1.1知识概述

本节课主要讲解SpringMVC中一个重要的注解@RequestParam。

##### 6.1.2视频详情



##### 6.1.3总结与补充

RequestParam注解

1. 作用：把请求中的指定名称的参数传递给控制器中的形参赋值

2. 属性

1. value：请求参数中的名称

2. required：请求参数中是否必须提供此参数，默认值是true，必须提供

##### 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 @RequestBody注解

##### 6.2.1知识概述

本节课主要讲解SpringMVC中一个重要的注解@RequestBody

##### 6.2.2视频详情



##### 6.2.3总结与补充

RequestBody注解

1. 作用：用于获取请求体的内容（注意：get方法不可以）

2. 属性

1. required：是否必须有请求体，默认值是true

##### 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 @PathVariable注解

##### 6.3.1知识概述

本节课主要讲解SpringMVC中一个重要的注解@PathVariable

##### 6.3.2视频详情



##### 6.3.3总结与补充

PathVariable注解

1. 作用：拥有绑定url中的占位符的。例如：url中有/delete/{id}，{id}就是占位符

2. 属性

1. value：指定url中的占位符名称

3. Restful风格的URL

1. 请求路径一样，可以根据不同的请求方式去执行后台的不同方法

2. restful风格的URL优点

1. 结构清晰

2. 符合标准

3. 易于理解

4. 扩展方便

##### 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 HiddentHttpMethodFilter过滤器

##### 6.4.1知识概述

本届主要讲解HiddentHttpMethodFilter过滤器的应用场景。

##### 6.4.2视频详情



##### 6.4.3总结与补充

无

##### 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

#### 6.5 @RequestHeader注解

##### 6.5.1知识概述

本节主要讲解@RequestHeader注解

##### 6.5.2视频详情



##### 6.5.3总结与补充

RequestHeader注解

1. 作用：获取指定请求头的值

2. 属性

1. value：请求头的名称

##### 6.5.4课堂提问与练习

无

##### 6.5.5习题答案

无

##### 6.5.6练习答案

无

##### 6.5.7视频缺陷

无

##### 6.5.8 视频扩展

无

#### 6.6 @CookieValue注解

##### 6.6.1知识概述

本节主要讲解@CookieValue注解。

##### 6.6.2视频详情



##### 6.6.3总结与补充

CookieValue注解

1. 作用：用于获取指定cookie的名称的值

2. 属性

1. value：cookie的名称

##### 6.6.4课堂提问与练习

无

##### 6.6..5习题答案

无

##### 6.6.6练习答案

无

##### 6.6.7视频缺陷

无

##### 6.6.8 视频扩展

无

#### 6.7 @ModelAttribute注解

##### 6.7.1知识概述

本节主要讲解@ModelAttribute注解

##### 6.7.2视频详情



##### 6.7.3总结与补充

ModelAttribute注解

1. 作用

1. 出现在方法上：表示当前方法会在控制器方法执行前线执行。

2. 出现在参数上：获取指定的数据给参数赋值。

2. 应用场景

1. 当提交表单数据不是完整的实体数据时，保证没有提交的字段使用数据库原来的数据。

##### 6.7.4课堂提问与练习

无

##### 6.7.5习题答案

无

##### 6.7.6练习答案

无

##### 6.7.7视频缺陷

无

##### 6.7.8 视频扩展

无

#### 6.8 @SessionAttributes注解

##### 6.8.1知识概述

本节课主要讲解@SessionAttributes注解。

##### 6.7.2视频详情



##### 6.7.3总结与补充

SessionAttributes注解

1. 作用：用于多次执行控制器方法间的参数共享

2. 属性

1. value：指定存入属性的名称

##### 6.7.4课堂提问与练习

无

##### 6.7.5习题答案

无

##### 6.7.6练习答案

无

##### 6.7.7视频缺陷

无

##### 6.7.8 视频扩展

无