导航

博客园
首 页
新随笔
联 系
订 阅 XML
管 理

<	2018年8月					>
日	一	二	三	四	五	六
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

公告

SpringMVC入门

目录

介绍

SpringMVC是一款Web MVC框架。 它跟Struts框架类似，是目前主流的Web MVC框架之一。

本文通过实例来介绍SpringMVC的入门知识。

实例

本文所写的实例是一个员工的CRUD demo。 用idea编写，基于maven，Web框架使用SpringMVC，视图采取Freemarker技术，数据库使用MySQL，用Hibernate4存储数据。

本文关于其他一些内容 如maven的pom文件内容，spring常规bean，事务，数据源的配置等不会详细描述，可自行下载代码查看

项目文件目录如下：

昵称：format丶
园龄：6年2个月
粉丝：384
关注：12
+加关注

搜索

找找看
谷歌搜索

常用链接

我的随笔
我的评论
我的参与
最新评论
我的标签

我的标签

java(23)
spring(14)
springmvc(11)
mybatis(4)
xml(3)
javascript(2)
json(2)
cache(2)
cglib(1)
classloader(1)
更多

随笔分类

cache(2)
css(1)
design pattern(1)
java(25)
javascript(3)
log(1)
maven
mybatis(4)
project
SpringMVC(12)
websocket(1)
xml(1)
挨踢生涯(1)
并发(1)

随笔档案

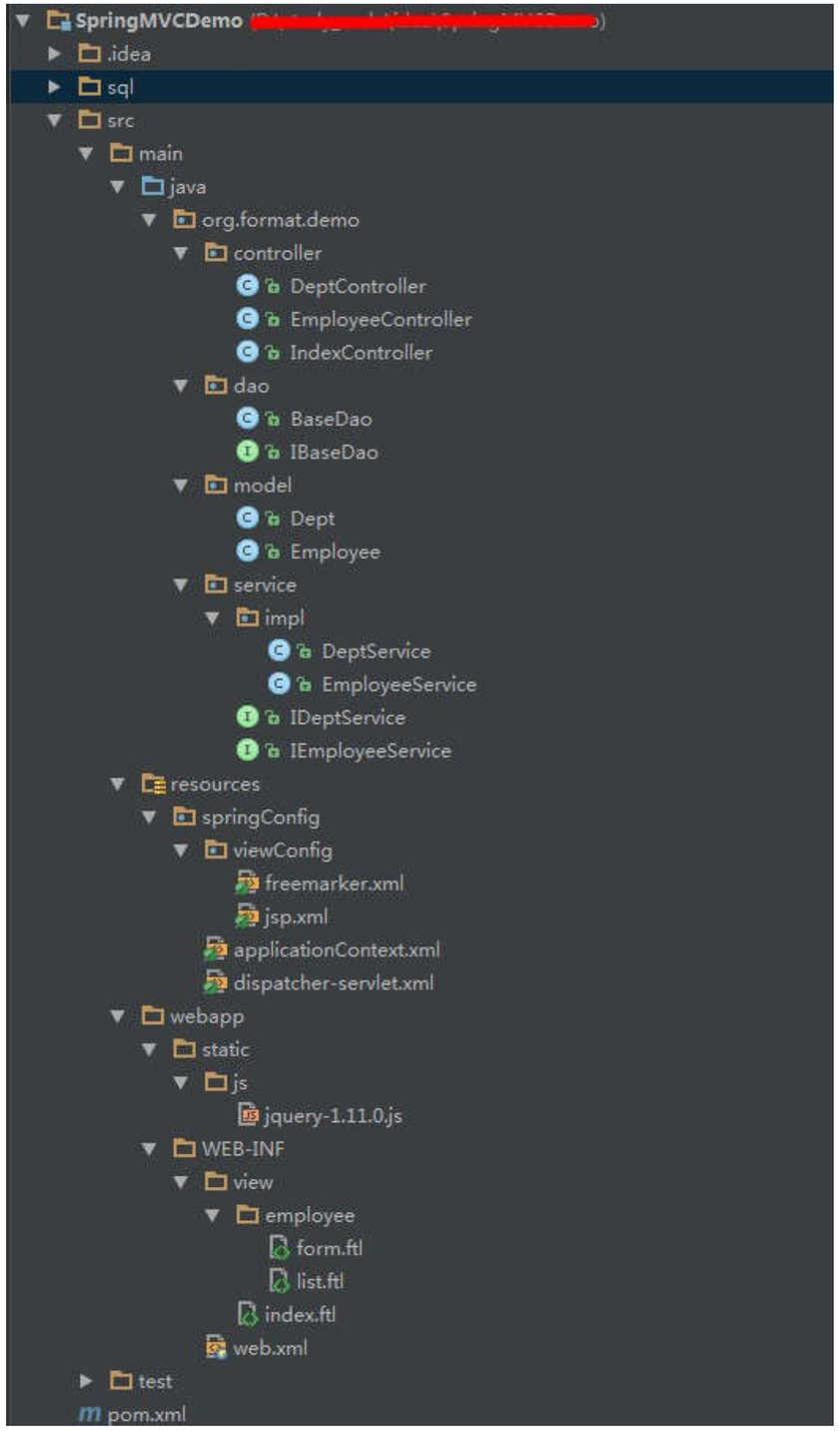
2015年8月 (1)
2014年12月 (2)
2014年11月 (2)
2014年9月 (3)
2014年8月 (3)
2014年7月 (4)
2014年6月 (9)
2014年5月 (5)
2014年4月 (2)
2014年3月 (1)
2014年2月 (2)
2014年1月 (1)

相册

image(66)

最新评论

1. Re:SpringMVC关于json、xml自动转换的原理研究[附带源码分析]
牛牛牛牛逼
--zombieG



首先在web.xml中配置入口servlet

```
<servlet>
    <servlet-name>dispatcher</servlet-name>
    <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
    <init-param>
        <param-name>contextConfigLocation</param-name>
        <param-value>classpath:springConfig/dispatcher-servlet.xml</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
</servlet>
<servlet-mapping>
    <servlet-name>dispatcher</servlet-name>
    <url-pattern>/</url-pattern>
</servlet-mapping>
```

注意有个初始参数 contextConfigLocation， 顾名思义 上下文配置文件路径。

还有另外一个注意点，这个servlet对应的url-pattern最好写成 "/"， 不要写成 "/*"。 至于为什么需要写成 "/" ， 具体请参考
http://www.cnblogs.com/fangjian0423/p/servletContainer-tomcat-urlPattern.html#springmvc

在dispatcher-servlet.xml中主要是配置springmvc的一些Controller的初始化，静态文件的映射策略，视图的配置等。

然后开始编写Controller类(类似Struts2中的Action)

首先来看一个主页的Controller:

```
@Controller
@RequestMapping("/")
public class IndexController {

    @RequestMapping
    public ModelAndView index() {
        ModelAndView view = new ModelAndView("index");
        view.addObject("welcome", "hello");
        return view;
    }

}
```

这里有2个注解： @Controller和@RequestMapping

8. 详解SpringMVC请求的时候是如何找到正确的Controller[附带源码分析](19977)
9. SpringMVC拦截器详解[附带源码分析](19169)
10. MyBatis拦截器原理探究(18680)

评论排行榜

1. 通过源码分析MyBatis的缓存(37)
2. Servlet容器Tomcat中web.xml中url-pattern的配置详解[附带源码分析](30)
3. SpringMVC关于json、xml自动转换的原理研究[附带源码分析](29)
4. SpringMVC源码分析系列(28)
5. SpringMVC入门(23)

推荐排行榜

1. SpringMVC源码分析系列(29)
2. Servlet容器Tomcat中web.xml中url-pattern的配置详解[附带源码分析](21)
3. SpringMVC入门(19)
4. SpringMVC关于json、xml自动转换的原理研究[附带源码分析](19)
5. 详解SpringMVC请求的时候是如何找到正确的Controller[附带源码分析](17)
6. 通过源码分析MyBatis的缓存(13)
7. SpringMVC核心分发器DispatcherServlet分析[附带源码分析](12)
8. 详解SpringMVC中Controller的方法中参数的工作原理[附带源码分析](12)
9. 最近状况的一点总结(2014年上半年总结)(10)
10. MyBatis拦截器原理探究(10)

@Controller注解就是表明这是一个Controller，且会被spring容器进行初始化。

dispatcher-servlet.xml中的扫描包配置语句：

```
<context:component-scan base-package="org.format.demo.controller" />
```

这条语句是扫描org.format.demo.controller下被@Controller(还有其他的如 @Component, @Service, @Repository)注解的那些类，并进行实例化。

@RequestMapping 顾名思义，就是请求映射。

我们看到@RequestMapping("/")中的"/"的意义就是contextPath后面的路径；也就是 http://host:port/contextPath 后面的路径。 **(这里不一定都要以"/"开头，比如 "/employee"， 我们可以写成 "employee")**

ModelAndView对象就是一个带模型的视图对象。 我们看到IndexController返回了1个index名称的ModelAndView对象。

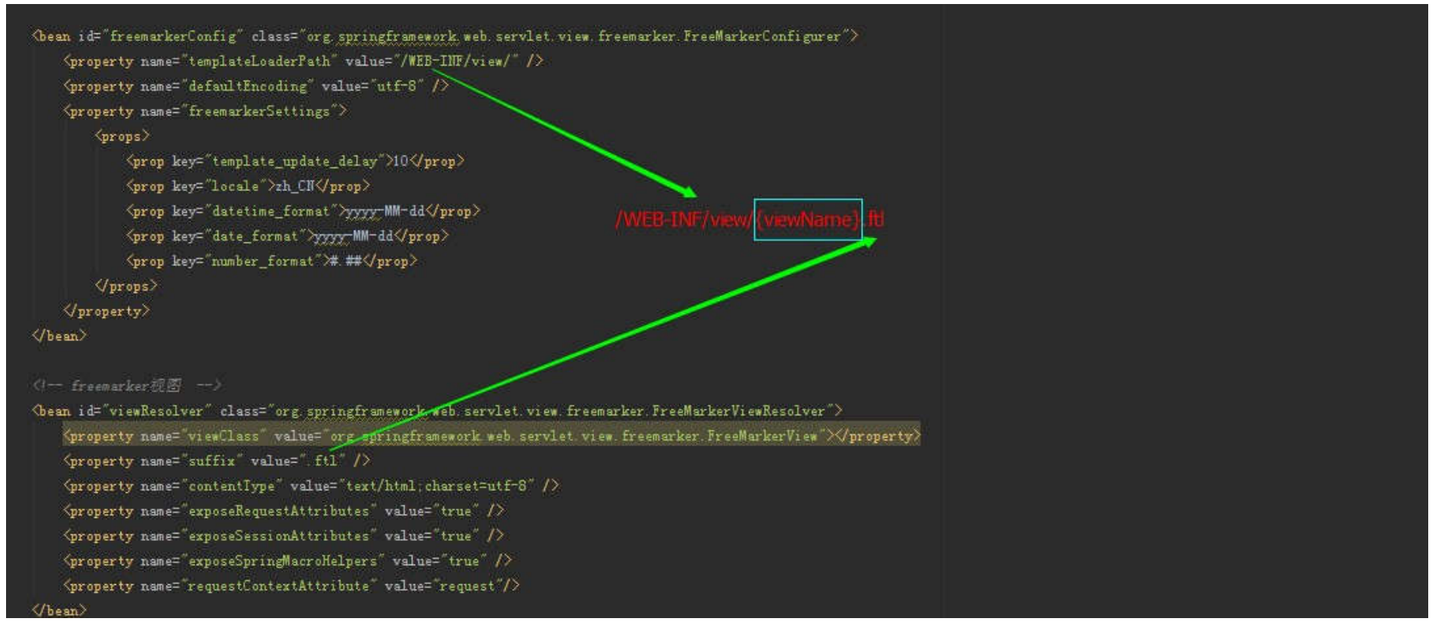
addObject对象就类似HttpServletRequest的setAttribute对象，也就是视图里面丢数据。 我们看到丢了一个key为welcome的对象。

最后的视图代码：

```
<!DOCTYPE html>
<html>
<head lang="en">
    <meta charset="UTF-8">
    <title></title>
</head>
<body>
    <h2>Welcome to user SpringMVC</h2>
    <h3>your welcome param: ${welcome}</h3>
</body>
</html>
```

我们看到用了el表达式得到了丢入的数据welcome，也就是hello。 因此，最终生成的页面及地址如下：

最后SpringMVC会通过配置文件中的视图对象拼接成最终的视图地址。



图中viewName也就是返回的ModelAndView中的viewName 即index。　因此，最终的视图路径为 /WEB-INF/view/index.ftl。

下面在来看一个员工操作的Controller代码：

```
@Controller
@RequestMapping(value = "/employee")
public class EmployeeController {

    @Autowired
    private IEmployeeService employeeService;

    @Autowired
    private IDeptService deptService;

    @RequestMapping
    public ModelAndView index() {
        ModelAndView view = new ModelAndView("employee/list");
        List<Employee> employees = employeeService.list();
        view.addObject("list", employees);
        return view;
    }
```

```
@RequestMapping(method = RequestMethod.POST, value = "/delete/{employeeId}")
@ResponseBody
public String delete(@PathVariable Integer employeeId) {
    employeeService.delete(employeeId);
    return "success";
}

@RequestMapping(method = RequestMethod.GET, value = "/add")
public ModelAndView add(ModelAndView view) {
    view.setViewName("employee/form");
    view.addObject("depts", deptService.listAll());
    return view;
}

@RequestMapping(method = RequestMethod.GET, value = "/detail/{employeeId}")
public ModelAndView detail(@PathVariable Integer employeeId, ModelAndView view) {
    view.setViewName("employee/form");
    view.addObject("employee", employeeService.getById(employeeId));
    view.addObject("depts", deptService.listAll());
    return view;
}

@RequestMapping(method = RequestMethod.POST, value = "/update")
    public String add(Employee employee) {
        if(employee.getDept().getId() == null) {
            employee.setDept(null);
        }
        employeeService.saveOrUpdate(employee);
        return "redirect:/employee/";
    }

}
```

这里多了几个新的内容：

1. @RequestMapping注解的作用位置

    @RequestMapping可以作用在类名上，也可以作用在方法上。 如果都有， 产生作用的路径是类名上的路径+方法上的路径。 比如EmployeeController的add方法，最终起作用的路径是 http://host:port/contextPath/employee/add

2. @RequestMapping注解的method参数

    method参数表示的HTTP请求的方式。常见的有GET,PUT,POST,DELETE等。若请求的方法与后台编写的方法不一致，会出现HTTP 405错误。

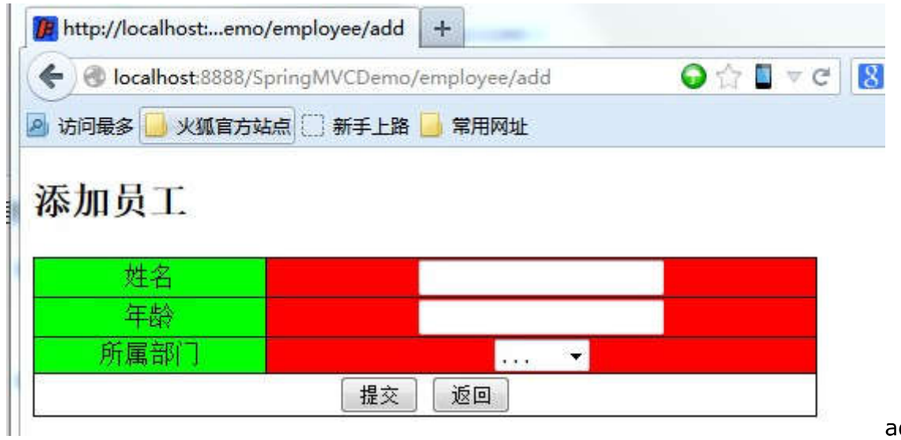
3. @PathVariable注解

这是一种基于RESTFUL的注解。我们看到detail方法的@RequestMapping的value值/detail/{employeeId}，参数中加入了一个@PathVariable employeeId。 这样起作用的路径就根据employee的Id，即每个员工都有独立的一个URI路径资源。 符合RESTFUL架构。

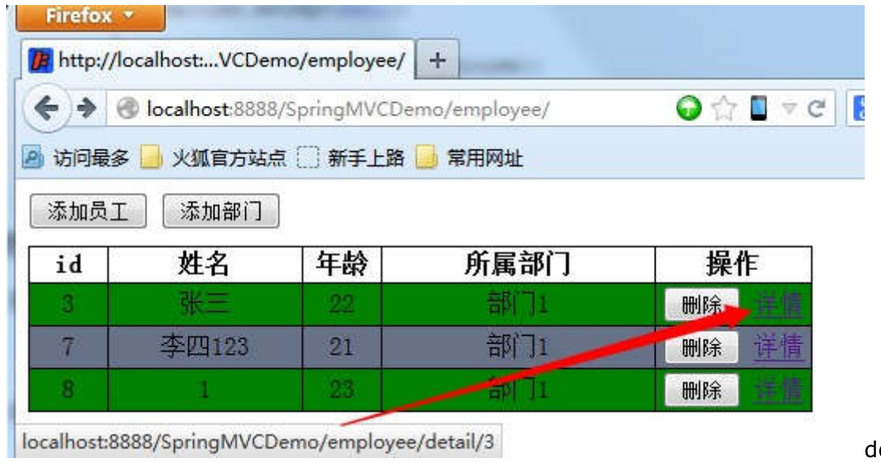
4. Controller的方法参数

Controller的方法访问非常灵活。 比如Employee有id,name,age等属性。 只要我们在前台传入name为id，name，age这3个参数，并且接受的方法有一个Employee对象参数，SpringMVC会自动把3个注入到这个对象中。 还有其他一些Integer，Long参数等，SpringMVC会默认帮我们自动转化。 同时参数也可以丢入一些HttpServletRequest, HttpServletResponse, HttpServletSession对象，SpringMVC会自动帮我们注入。 这点非常方便。

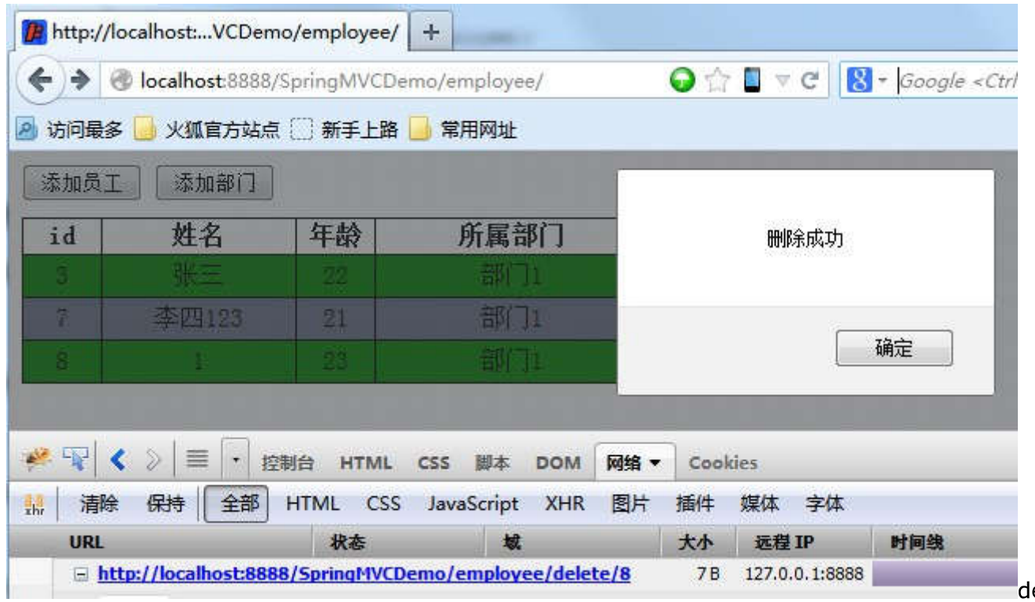
5. 不带参数基于方法的@RequestMapping会被当然基于类的@RequestMapping所作用的地址的默认进入的方法



add方法。 get请求



detail方法。 RESTFUL风格



delete方法。RESTFUL风格

总结

SpringMVC是一个Web MVC框架。 它的特点是轻便，与Spring无缝整合，上手简单。它的易用性、可扩展性、安全性均非常理想。

参考资料

http://www.ruanyifeng.com/blog/2011/09/restful.html

http://jinnianshilongnian.iteye.com/blog/1593441

代码下载地址： http://files.cnblogs.com/fangjian0423/SpringMVCDemo.zip

分类: java,SpringMVC

标签: java, spring, springmvc

好文要顶 关注我 收藏该文

format丶
关注 - 12
粉丝 - 384
+加关注

19 0

« 上一篇：Freemarker中Configuration的setClassForTemplateLoading方法参数问题
» 下一篇：Servlet容器Tomcat中web.xml中url-pattern的配置详解[附带源码分析]

posted on 2014-05-01 16:59 format丶 阅读(50048) 评论(23) 编辑 收藏

评论

#1楼

sptring3 吧

支持(0) 反对(0)

2014-05-01 21:33 | stillwater

#2楼[楼主]

@ stillwater
是的。 但是pom中写的是4.0.2版本。

支持(0) 反对(0)

2014-05-01 21:53 | format丶

#3楼[楼主]

@ 斯拉万岁
idea。
http://www.jetbrains.com/idea/

支持(0) 反对(0)

2014-05-02 13:45 | format丶

#4楼

楼主 您也用springmvc呀 我们能交流交流吗 您的QQ多少呢 我加吧 以后大家一起学习吧 谢谢

支持(0) 反对(0)

2014-05-04 08:58 | i surrender

#5楼[楼主]

@ i surrender
359941160。 共勉。

支持(0) 反对(0)

2014-05-04 09:49 | format丶

#6楼

博主您好！很想根据你的博文学习一下Spring的使用，能否导出一个Eclipse项目传我一份，或者供下载？我邮箱 hans_gis@163.com 多谢了！

支持(0) 反对(0)

2014-05-19 09:54 | 百折不回

#7楼[楼主]

@ 韩锁
你好。 我用的ide是idea，不是eclipse。强烈推荐你也用下。 本文最后有代码下载地址，如果你需要eclipse，我在idea里导出成eclipse给你。

支持(0) 反对(0)

2014-05-19 11:10 | format丶

#8楼

@ format丶
我正是需要eclipse工程，还麻烦你导出eclipse给我一份（hans_gis@163.com） 万分感谢！

支持(0) 反对(0)

2014-05-19 12:44 | 百折不回

#9楼[楼主]

@ 韩锁
已发了。 你进邮箱看下。

支持(0) 反对(0)

2014-05-19 13:03 | format丶

#10楼

分析系列不错啊！

支持(0) 反对(0)

2014-06-24 21:36 | 美洲象

#11楼[楼主]

@ 美洲象
谢谢支持

支持(0) 反对(0)

2014-06-24 21:50 | format丶

#12楼

楼主，我在使用SpringMVC的过程中遇到一个问题，比如我在进入一个页面之前是请求了一个admin/employee/detail/的路径后进行/employee/detail.jsp页面，然后在以后的请求中,在浏览器显示的路径里面，都会自动添加 admin/employee/ 这个路径，比如我请求的实际是book/add/，但在浏览器显示的却是admin/employee/book/add/,这个问题困惑了很久。目前的解决方案是，比如原来的请求路径是三层的，那必须在生成的页面的请求路径中添加 ../../，如果原来的请求是两层的，则必须添加../。不知道楼主知道这是哪里配置问题么

支持(0) 反对(0)

2014-06-26 10:54 | 叶汉城

#13楼[楼主]

@ 叶汉城
这是相对路径造成的问题。
进入的新页面路径建议使用 "/" 开头。

比如 book/add/ 写成 /book/add

支持(0) 反对(0)

2014-06-26 10:58 | format丶

#14楼

@ format丶

引用

@叶汉城这是相对路径造成的问题。
进入的新页面路径建议使用 "/" 开头。

比如 book/add/ 写成 /book/add

即使这样改还是一样不行的，不起效果！奇怪的是，接触几个项目基本都是这样配置的，对比不出有什么差别、但其它项目都是别人搭的，唯独这个项目自己搭的，出了这么奇怪的问题。我QQ402418306，如果可以的话，麻烦加下QQ，我跟你请教下

支持(0) 反对(0)

2014-06-26 14:51 | 叶汉城

#15楼

最近也在学java，我是c#转过来的，看得懂，不过配置感觉挺麻烦。
建议咱们建个群吧，相互学习啊。。我加你QQ

支持(0) 反对(0)

2014-06-27 16:28 | vians

#16楼

呵呵，请问你的黑色主题怎么设置的呢？我安网上的设置后总是看起来很别扭！！

支持(0) 反对(0)

2015-03-31 16:45 | 没得不安逸

#17楼

博主您好！可否导出一份eclipse的发到我的邮箱，最近我在学习springMVC。我的邮箱是1107127710@qq.com

支持(0) 反对(0)

2016-04-16 11:51 | 不一样吗

#18楼

博主你好，为了访问静态资源配置了

```
<mvc: default-servlet-handler />
```

之后，须配置

```
<mvc: annotation-driven />
```

才能访问Controller.请问这是什么原因?

支持(0) 反对(0)

2016-07-08 23:42 | stringang

#19楼

这不是 restful 风格的吧~rest 风格api 没有动词

支持(0) 反对(0)

2016-09-18 17:01 | HelloBobi

#20楼

总结的到位 受教了

支持(0) 反对(0)

2017-01-02 20:09 | 酒是没有解药的毒

#21楼

学习了，springMVC

支持(0) 反对(0)

2017-10-16 20:29 | 小妖，快跑我断后

#22楼

不能运行，报错：
通配符的匹配很全面, 但无法找到元素 'context:component-scan' 的声明。

支持(0) 反对(0)

2017-11-05 12:44 | ajupiter

#23楼[楼主]

@ TopPlayer
应该xml中没有加上对应的schema吧。
可以参考：https://github.com/fangjian0423/SpringMVCSourceCodeLearn 里的代码

支持(0) 反对(0)

2017-11-09 17:29 | format丶