Spring mvc 学习笔记

1. spring 3.1及更高版本支持通过注解的方式进行配置。
2. 处理器的定义：@Controller、@RequestMapping

|  |
| --- |
| package cn.javass.chapter6.web.controller;  @Controller  @RequestMapping(value="/user") //①处理器的通用映射前缀  public class HelloWorldController2 {  @RequestMapping(value = "/hello2") //②相对于①处的映射进行窄化  public ModelAndView helloWorld() {  //省略实现  }  } |

推荐使用这种方式声明处理器，它和我们的@Service、@Repository很好的对应了我们常见的三层开发架构的组件。

1. 请求映射
   1. URL路径映射
      1. 普通URL映射：

@RequestMapping(value={"/test1", "/user/create"})：多个URL路径可以映射到同一个处理器的功能处理方法。

* + 1. URI模板模式映射

@RequestMapping(value="/users/{userId}")：{×××}占位符， 请求的URL可以是 “/users/123456”或“/users/abcd”，可通过@PathVariable提取URI模板模式中的{×××}中的×××变量。

* + 1. ANT风格的URL路径映射

@RequestMapping(value="/users/\*\*")：可以匹配“/users/abc/abc”，但“/users/123”将会被【URI模板模式映射中的“/users/{userId}”模式优先映射到】。

* + 1. 正则表达式风格的URL路径映射

@RequestMapping(value="/products/{categoryCode:\\d+}-{pageNumber:\\d+}")：可以匹配“/products/123-1”，但不能匹配“/products/abc-1”，这样可以设计更加严格的规则。

* 1. 请求方法映射限定
     1. 只接受POST方式提交：@RequestMapping(value="/create", method = RequestMethod.POST)
     2. 同时支持POST/GET提交：@RequestMapping(value="/methodOr", method = {RequestMethod.POST, RequestMethod.GET})
  2. 请求参数映射限定
     1. 请求中包含参数

@RequestMapping(params="create", method=RequestMethod.GET) ：表示请求中有“create”的参数名且请求方法为“GET”即可匹配，如可匹配的请求URL“http://×××/parameter1?create”。

* + 1. 请求不含特定参数

@RequestMapping(params="!create", method=RequestMethod.GET)：表示请求中没有“create”参数名且请求方法为“GET”即可匹配，如可匹配的请求URL“http://×××/parameter1?abc”。

* + 1. 请求特定参数为特定值

@RequestMapping(params="submitFlag=create", method=RequestMethod.POST)：表示请求中有“submitFlag=create”请求参数且请求方法为“POST”即可匹配

* 1. 请求头映射限定：略

1. Xxx