Spring MVC接收请求参数的方法

时间 2014-06-19 17:23:04  [IT社区推荐资讯](http://www.tuicool.com/sites/q6BN7f)

原文  [http://itindex.net/detail/50082-spring-springmvc-方法](http://itindex.net/detail/50082-spring-springmvc-%E6%96%B9%E6%B3%95?utm_source=tuicool&utm_medium=referral)

主题 [Spring MVC](http://www.tuicool.com/topics/11070101)

     控制器的请求处理方法的签名可以将任何事物作为参数，我们可以将HttpServletRequest、HttpServletResponse、 String或者数字参数传递进来,这些参数可以对应请求中的查询参数、cookie值、HTTP请求头的值或其他一些可能的选项。

     下面一些例子来说明

1. 使用**@RequestParam**绑定请求参数值
2. @RequestMapping(value="/method1")
3. **public** String method1(
4. @RequestParam(value="userName",required = **false**) String userName,
5. @RequestParam("pwd") String pwd){
6. **return** "methodSignature";

}

1. 使用**@CookieValue**绑定请求中的Cookie值
2. @RequestMapping(**value**="/method2")
3. **public** String method2(
4. @CookieValue(**value**="sessionId",required=**false**)String sessionId){
6. **return** "methodSignature";

}

1. 使用**@RequestHeader**绑定请求报文头的属性值
2. @RequestMapping(value="/method3")
3. **public** String method3(
4. @RequestHeader("Accept-Encoding") String encoding,
5. @RequestHeader("Keep-Alive") **long** keepAlive){
6. **return** "methodSignature";

}

1. 使用**@PathVariabl**获取路径中传递参数
2. **public** ModelAndView helloWorld(
3. @PathVariable String id,
4. @PathVariable String str) {
5. System.**out**.println(id);
6. System.**out**.println(str);
7. **return** **new** ModelAndView( " /helloWorld " );

}

1. 使用表单对象绑定请求参数值
2. @RequestMapping(**value**="/method4")
3. **public** String method4(User user){
4. **return** "methodSignature";

}

1. 使用**Servlet API**对象作为入参
   * 同时使用**HttpServletRequest、HttpServletResponse、HttpSession**
   * @RequestMapping(**value**="method5")
   * **public** **void** method5(
   * HttpServletRequest request,
   * HttpServletResponse response,
   * HttpSession session){
   * String userName = WebUtils.findParameterValue(request, "userName");
   * response.addCookie(**new** Cookie("userName",userName));

}

* + 使用**Servlet原生API**接口，如WebRequest和NativeWebRequest等
  + @RequestMapping(**value**="method6")
  + **public** String method6(WebRequest request){
  + String userName = request.getParameter("userName");
  + **return** "methodSignature";

}

* + 使用**IO**对象作为入参    
    如java.io.InputStream/java.io.Reader 及java.io.OutputStream/java.io.Writer
  + @RequestMapping(value="method7")
  + **public** **void** method7(OutputStream os) **throws** IOException{
  + Resource res = **new** ClassPathResource("/image.jpg");*//读取类路径下的图片文件*
  + FileCopyUtils.copy(res.getInputStream(), os);

}

* + 使用其他类型的参数   
    **java.util.Locale/java.security.Principal**  
    也可以通过Servlet的**HttpServletRequest的getLocale()及getUserPrincipal()**
  + @RequestMapping(**value**="method8")
  + **public** String method8(Locale locale,Principal principal){
  + **return** "methodSignature";

}

Spring MVC接收请求参数的方法2

通过@PathVariabl注解获取路径中传递参数

转载请注明出处：[**springmvc请求接收参数的几种方法**](http://www.zuidaima.com/share/1751862044773376.htm)

代码下载地址：<http://www.zuidaima.com/share/1751862044773376.htm>  
  
Java

1. @RequestMapping(value= " /{id}/{str} " )
2. **public** ModelAndView helloWorld(@PathVariable String id, @PathVariable String str) {
3. System.out.println(id);
4. System.out.println(str);
5. **return** **new** ModelAndView( " /helloWorld " );
6. }

用@ModelAttribute注解获取POST请求的FORM表单数据  
JSP

1. <form method="post" action="hao.do">
2. a: <input id="a" type="text" name="a"/>
3. b: <input id="b" type="text" name="b"/>
4. <input type="submit" value="Submit" />
5. </form>

 Java pojo

1. **public** **class** Pojo{
2. **private** String a;
3. **private** **int** b;

 JAVA controller

1. @RequestMapping(method= RequestMethod.POST)
2. **public** String processSubmit(@ModelAttribute( " pojo " ) Pojo pojo) {
3. **return** " helloWorld " ;
4. }

直接用HttpServletRequest获取  
JAVA

1. @RequestMapping(method= RequestMethod.GET)
2. **public** String get(HttpServletRequest request, HttpServletResponse response) {
3. System.out.println(request.getParameter( " a " ));
4. **return** " helloWorld " ;
5. }

用注解@RequestParam绑定请求参数a到变量a  
当请求参数a不存在时会有异常发生,可以通过设置属性required=false解决,  
例如: @RequestParam(value="a", required=false)  
JAVA

1. @RequestMapping(value= " /requestParam " , method= RequestMethod.GET)
2. **public** String setupForm(@RequestParam( " a " ) String a, ModelMap model) {
3. System.out.println(a);
4. **return** " helloWorld " ;}