# [springMVC:将controller中数据传递到jsp页面](http://blog.csdn.net/sunshine__me/article/details/49494545)

1> 将方法的返回值该为ModelAndView在返回时，将数据存储在ModelAndView对象中如：

newModelAndView("/WEBINF/jsp/showData.jsp","message",message)

其中第一个参数为url,第二个参数为要传递的数据的key,第三个参数为数据对象。在这里要注意的是 数据是默认被存放在request中的。

示例:

|  |
| --- |
| @RequestMapping(value="/mad/showData\_1.do")  public ModelAndView showData\_1(){  String message = "这个是要传递的数据";  /\*其中第一个参数为url,第二个参数为要传递的数据的key,第三个参数为数据对象。在这里要注意的是:数据是默认被存放在request中的。\*/  return new ModelAndView("/WEB-INF/jsp/showData.jsp","message",message);  } |

前台页面获取方式:

request:${requestScope.message}

2> 可以在类的前面添加注解

@SessionAttributes({“message”,”user”})

这个注解可以设置对应的model中参数也会在session中存储一份。该注解中的参数为一个集合，可以写多个，如上面的例子，其中message和user都是存储的数据的key.

示例程序：

|  |
| --- |
| @SessionAttributes({"message","user"}) //modelAndView中的对应的数据也会在session中存储一份 |

页面获取：

|  |
| --- |
| session:${sessionScope.message}<br/> |

3 > 数据modelAndView返回一个集合

该处理方式和上面的处理方式一直，因为modelAndView接受的数据类型是Object的，集合也是一样的处理方式

示例:

|  |
| --- |
| //使用modelAndView对象将数据传递到前台。  //传递多个参数（不同类型的）  @RequestMapping(value="/mad/showData\_2.do")  public ModelAndView showData\_2(){  System.out.println("showData\_2");  String message = "这个是要传递的数据";  User user = new User("张三", 12, new Date());  List<User> us= new ArrayList<User>();  us.add(new User("张三", 12, new Date()));  us.add(new User("张四", 13, new Date()));  us.add(new User("张五", 14, new Date()));  ModelAndView mad = new ModelAndView("/WEB-INF/jsp/showData.jsp");  //将数据存入modelMap  mad.addObject("message", message);  mad.addObject(user);//默认为类名的首字母小写  mad.addObject("users", us);  return mad;  } |

前台页面获取方式:

|  |
| --- |
| request:${requestScope.message}<br/>  <c:forEach items="${requestScope.users }"var="u">  ${u.name }-${u.age }-${u.birth }<br/>  </c:forEach> |

4>使用modelAndView传递多个参数。

可以通过ModelAndView的mad.addObject(“message”,message);方法设置参数。

该方法中第一个参数为数据的key，第二个参数为数据对象。

也可以使用mad.addObject(user);方法

该方法中数据的参数为数据对象，数据的key为该对象的类的类名

（其中首字母小写）。

5>使用ModelMap传递多个数据到jsp中。

在方法的参数列表中添加形参 ModelMap map,spring 会自动创建ModelMap对象。

然后调用map的put(key,value)或者addAttribute(key,value)将数据放入map中，spring会自动将数据存入request。

示例:

|  |
| --- |
| //使用modelMap对象将数据传递到前台。  //传递多个参数（不同类型的）  @RequestMapping(value="/mad/showData\_3.do")  public String showData\_3(ModelMap map){  System.out.println("showData\_3");  String message = "这个是要传递的数据";  User user = new User("张三", 12, new Date());  List<User> us= new ArrayList<User>();  us.add(new User("张三", 12, new Date()));  us.add(new User("张四", 13, new Date()));  us.add(new User("张五", 14, new Date()));  //将数据存入modelMap  map.put("message", message);  map.addAttribute("user", user);  map.put("users", us);  return"/WEB-INF/jsp/showData.jsp";  } |

页面调用:

|  |
| --- |
| request:${requestScope.message}<br/>  session:${sessionScope.message }<br/>  application:${applicationScope.message}<br/>  <hr/>  <h1>ModelMap中的数据</h1>  ${requestScope.message}<br/>  ${requestScope.user.name}<br/>  <p>列表</p>  <c:forEach items="${requestScope.users}" var="u">  ${u.name }-${u.age }-${u.birth }<br/>  </c:forEach> |

原文地址: http://blog.csdn.net/sunshine\_\_me/article/details/49494545

**附:** SpringMVC的几种返回方式

http://blog.csdn.net/sunhuwh/article/details/41727257