# **[spring](http://lib.csdn.net/base/17" \o "Java EE知识库" \t "http://blog.csdn.net/z69183787/article/details/_blank)**接收请求参数:

## 1,使用HttpServletRequest获取

**Java代码  IMG_256**

1. @RequestMapping("/login.do")
2. **public** String login(HttpServletRequest request){
3. String name = request.getParameter("name")
4. String pass = request.getParameter("pass")
5. }

## 2,Spring会自动将表单参数注入到方法参数，和表单的name属性保持一致。和Struts2一样

**Java代码  IMG_257**

1. @RequestMapping("/login.do")
2. **public** String login(HttpServletRequest request,
3. String name,
4. @RequestParam("pass")String password) // 表单属性是pass,用变量password接收
5. {
6. syso(name);
7. syso(password)
8. }

## 3,自动注入Bean属性

**Java代码  IMG_258**

1. <form action="login.do">
2. 用户名：<input name="name"/>
3. 密码：<input name="pass"/>
4. <input type="submit" value="登陆">
5. </form>
7. //封装的User类
8. **public** **class** User{
9. **private** String name;
10. **private** String pass;
11. }

**Java代码  IMG_259**

1. @RequestMapping("/login.do")
2. **public** String login(User user)
3. {
4. syso(user.getName());
5. syso(user.getPass());
6. }

# 向页面传值：

**当Controller组件处理后，向jsp页面传值，**

**1，使用HttpServletRequest 和 Session  然后setAttribute()，就和Servlet中一样**

**2，使用ModelAndView对象**

**3，使用ModelMap对象**

**4，使用@ModelAttribute注解**

## 使用HttpServletRequest

@RequestMapping("/login")

**public** String login(HttpServletRequest request){

request.setAttribute("theValue", "Hello request");

**return** "result";

}

## 使用ModelAndView对象

**Model数据会利用HttpServletRequest的Attribute传值到success.jsp中**

**Java代码  IMG_260**

1. @RequestMapping("/login.do")
2. **public** ModelAndView  login(String name,String pass){
3. User user = userService.login(name,pwd);
4. Map<String,Object> data = **new** HashMap<String,Object>();
5. data.put("user",user);
6. **return** **new** ModelAndView("success",data);
7. }

## 使用ModelMap:

**ModelMap数据会利用HttpServletRequest的Attribute传值到success.jsp中**

**Java代码  IMG_261**

1. @RequestMapping("/login.do")
2. **public**　String login(String name,String pass ,ModelMap model){
3. User user  = userService.login(name,pwd);
4. model.addAttribute("user",user);
5. model.put("name",name);
6. **return** "success";
7. }

## 使用@ModelAttribute示例

**在Controller方法的参数部分或Bean属性方法上使用**

**@ModelAttribute数据会利用HttpServletRequest的Attribute传值到success.jsp中**

**Java代码  IMG_262**

1. @RequestMapping("/login.do")
2. **public** String login(@ModelAttribute("user") User user){
3. //TODO
4. **return** "success";
5. }
7. @ModelAttribute("name")
8. **public** String getName(){
9. **return** name;
10. }

## 使用Session：

**可以利用HttpServletReequest的getSession()方法**

**Java代码  IMG_263**

1. @RequestMapping("/login.do")
2. **public** String login(String name,String pwd
3. ModelMap model,HttpServletRequest request){
4. User user = serService.login(name,pwd);
5. HttpSession session = request.getSession();
6. session.setAttribute("user",user);
7. model.addAttribute("user",user);
8. **return** "success";
9. }

# 重定向：

**Spring MVC 默认采用的是转发来定位视图，如果要使用重定向，可以如下操作**

## 1，使用RedirectView

**Java代码  IMG_264**

1. **public** ModelAndView login(){
2. RedirectView view = **new** RedirectView("regirst.do");
3. **return** **new** ModelAndView(view);
4. }

## 2，使用redirect:前缀

**工作中常用的方法：**

**Java代码  IMG_265**

1. **public** String login(){
2. //TODO
3. **return** "redirect:regirst.do";
4. }

# Note：

Jsp页面中获取参数请参照示例代码

在springMVC项目中使用$从jsp页面中获取参数与在Struts2中有所不同，具体参照示例项目