其实就是三种：

1. **直接把表单的参数写在Controller相应的方法的形参中**
2. **通过HttpServletRequest接收**
3. **通过一个bean来接收（JSON转bean）**

1. 前台主要代码

**[html]** [view plain](https://blog.csdn.net/jiuba5/article/details/52136828) [copy](https://blog.csdn.net/jiuba5/article/details/52136828)

1. **<form>**
2. **<table>**
3. **<tr>**
4. **<td>**账号**</td>**
5. **<td>**
6. **<input** type="text" name="userName"**>**
7. **</td>**
8. **</tr>**
9. **<tr>**
10. **<td>**密码**</td>**
11. **<td>**
12. **<input** type="password" name="password"**>**
13. **</td>**
14. **</tr>**
15. **<tr>**
16. **<td>** **</td>**
17. **<td>**
18. **<input** type="button" value="提交" onclick="addUser()"**>**
19. **</td>**
20. **</tr>**
21. **</table>**
22. **</form>**

2. 接收方式1------直接把表单的参数写在Controller相应的方法的形参中

**[html]** [view plain](https://blog.csdn.net/jiuba5/article/details/52136828) [copy](https://blog.csdn.net/jiuba5/article/details/52136828)

1. @RequestMapping("/addUser1")
2. public String addUser1(String userName,String password) {
3. System.out.println("userName is:"+userName);
4. System.out.println("password is:"+password);
5. return "/user/success";
6. }

3. 接收方式2-----通过HttpServletRequest接收

**[html]** [view plain](https://blog.csdn.net/jiuba5/article/details/52136828) [copy](https://blog.csdn.net/jiuba5/article/details/52136828)

1. @RequestMapping("/addUser2")
2. public String addUser2(HttpServletRequest request) {
3. String userName = request.getParameter("userName");
4. String password = request.getParameter("password");
5. System.out.println("userName is:"+userName);
6. System.out.println("password is:"+password);
7. return "/user/success";
8. }

4. 通过一个bean来接收

     （1）建立一个和表单参数对应的bean

**[html]** [view plain](https://blog.csdn.net/jiuba5/article/details/52136828) [copy](https://blog.csdn.net/jiuba5/article/details/52136828)

1. public class User {
3. private String userName;
5. private String password;
7. public String getUserName() {
8. return userName;
9. }
11. public void setUserName(String userName) {
12. this.userName = userName;
13. }
15. public String getPassword() {
16. return password;
17. }
19. public void setPassword(String password) {
20. this.password = password;
21. }
22. }

    2)通过这个bean来封装接收的参数

**[java]** [view plain](https://blog.csdn.net/jiuba5/article/details/52136828) [copy](https://blog.csdn.net/jiuba5/article/details/52136828)

1. @RequestMapping("/addUser3")
2. **public** String addUser3(User user) {
3. System.out.println("userName is:"+user.getUserName());
4. System.out.println("password is:"+user.getPassword());
5. **return** "/user/success";
6. }

5. 通过json 数据来接收参数

**[html]** [view plain](https://blog.csdn.net/jiuba5/article/details/52136828) [copy](https://blog.csdn.net/jiuba5/article/details/52136828)

1. **<**%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%**>**
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"**>**
4. **<html>**
5. **<head>**
6. **<meta** http-equiv="Content-Type" content="text/html; charset=UTF-8"**>**
7. **<title>**Add User**</title>**
9. **<script** type="text/javascript" src="${pageContext.request.contextPath}/resource/script/jquery-1.9.1.min.js"**></script>**
10. **<script** type="text/javascript"**>**
11. $(document).ready(function(){
12. $("#button\_submit").click(function(){
13. var name = $("#userName").val();
14. var pass = $("#password").val();

17. var user = {userName:name,password:pass};//拼装成json格式
19. $.ajax({
20. type:"POST",
21. url:"${pageContext.request.contextPath}/user/addUser4",
22. data:user,
23. success:function(data){
24. alert("成功");
25. },
26. error:function(e) {
27. alert("出错："+e);
28. }
29. });
30. });
31. });
32. **</script>**
34. **</head>**
35. **<body>**
36. **<form>**
37. **<table>**
38. **<tr>**
39. **<td>**账号**</td>**
40. **<td>**
41. **<input** type="text" id="userName" name="userName"**>**
42. **</td>**
43. **</tr>**
44. **<tr>**
45. **<td>**密码**</td>**
46. **<td>**
47. **<input** type="password" id="password" name="password"**>**
48. **</td>**
49. **</tr>**
50. **<tr>**
51. **<td>** **</td>**
52. **<td>**
53. **<input** type="button" id="button\_submit" value="提交"**>**
54. **</td>**
55. **</tr>**
56. **</table>**
57. **</form>**
58. **</body>**
59. **</html>**

  依然可以用Bean来接收json数据

**[html]** [view plain](https://blog.csdn.net/jiuba5/article/details/52136828) [copy](https://blog.csdn.net/jiuba5/article/details/52136828)

1. @RequestMapping("/addUser4")
2. public String addUser4(User user) {
3. System.out.println("userName is:"+user.getUserName());
4. System.out.println("password is:"+user.getPassword());
5. return "/user/success";
6. }

6、 使用jQuery的serializeArray()方法序列化表单元素

       如果表单元素很多，手工拼装成Json数据很麻烦，可以使用JQuery提供的SerializeArray()方法序列化表单元素，返回Json数据结构数据。

**[html]** [view plain](https://blog.csdn.net/jiuba5/article/details/52136828) [copy](https://blog.csdn.net/jiuba5/article/details/52136828)

1. **<**%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%**>**
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"**>**
4. **<html>**
5. **<head>**
6. **<meta** http-equiv="Content-Type" content="text/html; charset=UTF-8"**>**
7. **<title>**Add User**</title>**
9. **<script** type="text/javascript" src="${pageContext.request.contextPath}/resource/script/jquery-1.9.1.min.js"**></script>**
10. **<script** type="text/javascript"**>**
11. $(document).ready(function(){
12. $("#button\_submit").click(function(){
14. //序列化表单元素，返回json数据
15. var params = $("#userForm").serializeArray();
17. //也可以把表单之外的元素按照name value的格式存进来
18. //params.push({name:"hello",value:"man"});
20. $.ajax({
21. type:"POST",
22. url:"${pageContext.request.contextPath}/user/addUser5",
23. data:params,
24. success:function(data){
25. alert("成功");
26. },
27. error:function(e) {
28. alert("出错："+e);
29. }
30. });
31. });
32. });
33. **</script>**
35. **</head>**
36. **<body>**
37. **<form** id="userForm"**>**
38. **<table>**
39. **<tr>**
40. **<td>**账号**</td>**
41. **<td>**
42. **<input** type="text" id="userName" name="userName"**>**
43. **</td>**
44. **</tr>**
45. **<tr>**
46. **<td>**密码**</td>**
47. **<td>**
48. **<input** type="password" id="password" name="password"**>**
49. **</td>**
50. **</tr>**
51. **<tr>**
52. **<td>** **</td>**
53. **<td>**
54. **<input** type="button" id="button\_submit" value="提交"**>**
55. **</td>**
56. **</tr>**
57. **</table>**
58. **</form>**
59. **</body>**
60. **</html>**

依然可以使用bean来接收json数据

**[java]** [view plain](https://blog.csdn.net/jiuba5/article/details/52136828) [copy](https://blog.csdn.net/jiuba5/article/details/52136828)

1. @RequestMapping("/addUser5")
2. **public** String addUser5(User user) {
3. System.out.println("userName is:"+user.getUserName());
4. System.out.println("password is:"+user.getPassword());
5. **return** "/user/success";
6. }