**[Ajax与Controller数据交互](https://www.cnblogs.com/kasi/p/8443289.html)**

**1.概念**

在ssm框架下实现Ajax与Controller之间各种类型数据的数据交互,从Ajax传值和Controller返回值两个方面理解

**1.1 从Ajax → Controller**

* 无论Ajax以什么形式发送数据,在Controller中的接收方法都没有差异

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 对象 | 操作 | key/value类型 | JSON | 序列化表单 |
| Ajax | 发送 | data:"name="+name | data:{"name":name} | data:$("#form").serialize() |
| Controller | 接收 | public void receive(String name) / (User user) | | |

**1.2 从Controller → Ajax**

* Controller返回Json类型的数据需要加@ResponseBody注解
* Ajax的返回值data可以是任何名称

|  |  |  |  |
| --- | --- | --- | --- |
| 对象 | 操作 | JSON | |
| pojo类型 | map类型 |
| Controller | 发送 | return Msg; | return map; |
| Ajax | 接收 | success:function(data){ alert(data.result) | |

**2. 实例说明**

* JSP表单
* <form id="formId">
* 姓名:<input type="text" name="name" id="name"><br/>
* 年龄:<input type="password" name="pass" id="pass"><br/>
* 性别:<input type="radio" name="sex" value="m">男
* <input type="radio" name="sex" value="f">女<br/>
* 爱好:<input type="checkbox" name="hobby" value="basketball">篮球
* <input type="checkbox" name="hobby" value="football">足球
* <input type="checkbox" name="hobby" value="pingpang">乒乓球<br/>
* 地址:<input type="text" name="address" id="address"><br/>
* <input type="button" value="提交" id="sendTo">

</form>

**2.1 Ajax传值**

* 方式一:key/value传值
  + 普通类型传递需要注意复选框如果要和其他值一起传最好将其局部序列化
  + <script type="text/javascript">
  + $("#sendTo").click(function () {
  + //获取值
  + var name = $("#name").val();
  + var age = $("#age").val();
  + var sex = $("input[type='radio']").val();
  + var hobby = $("input[name='hobby']:checked").serialize(); //此处为复选框,用序列化的方式传递
  + var address = $("#address").val();
  + $.ajax({
  + url:"toServer.do",
  + type:"post",
  + //注意序列化的值一定要放在最前面,并且不需要头部变量,不然获取的值得格式会有问题
  + data:hobby+"&name="+name+"&age="+age+"&sex="+sex+"&address="+address,
  + dataType:"json",
  + success:function (data) {
  + // alert(data.result);
  + alert(data.result);
  + }
  + })
  + })

</script>

* 方式二:JSON传值
  + JSON与key/value只有在data的数据格式不一样(有些资料说需要加contentType:"application/json; charset=utf-8",但我没加也可以传值
  + //此处如果加入序列化后的复选框就传不过去,暂时还没弄清楚怎么将序列化后的值与json数据一起传,如果只是传复选框可以用"data:hobby,"这种方式

data:{"name":name,"age":age,"sex":sex,"address":address},

* 方式三:序列化(数据量大时推荐使用)
  + 在遇到复杂的表单时很有用
  + <script type="text/javascript">
  + $("#sendTo").click(function () {
  + $.ajax({
  + url:"toServer.do",
  + type:"post",
  + data:$("#formId").serialize(), //序列化表单
  + dataType:"json",
  + success:function (data) { //返回值data为{"result":"提交成功"}
  + alert(data.result);
  + }
  + })
  + }

</script>

**2.2 Controller返回值**

Controller接收值常用的就两种,一种是springmvc的参数绑定,另一种为JavaBean类型接收

* 方式一:map集合返回
  + 注意:复选框hobby值的形式取决于userbean中的hobby类型,String类型时传过来的值是逗号隔开的
  + @RequestMapping("/toServer.do")
  + @ResponseBody
  + public Map<String,String> sendString(User user){ //user是与页面参数对应的JavaBean
  + //map集合用来存放返回值
  + Map<String,String> map = new HashMap<String, String>();
  + if(user != null) {
  + map.put("result","提交成功");
  + }else {
  + map.put("result","提交失败");
  + }
  + return map;

}

* 方式二:Pojo返回
  + 先定义一个用于返回数据的Pojo
  + public class Msg {
  + private int code; //返回100表示成功,200表示失败
  + private String msg; //返回提示信息
  + private Map<String,Object> extend = new HashMap<String,Object>(); //用户返回给浏览器的数据
  + public static Msg success() {
  + Msg result = new Msg();
  + result.setCode(100);
  + result.setMsg("处理成功");
  + return result;
  + }
  + public static Msg fail() {
  + Msg result = new Msg();
  + result.setCode(200);
  + result.setMsg("处理失败");
  + return result;
  + }
  + public Msg add(String key,Object value) {
  + this.getExtend().put(key, value);
  + return this;
  + }

//get&set方法

* + Controller
  + @RequestMapping("/toServer.do")
  + @ResponseBody
  + public Msg sendString(User user){
  + System.out.println(user.toString());
  + if(user != null) {
  + return Msg.success();
  + }else {
  + return Msg.fail();
  + }
  + }
  + // ajax的success:function(data),data的返回值为{"code":100,"result":"成功"}
  + /\*此pojo可以使用return Msg.success.add("user",user)的方式向ajax返回实体对象

{"code":100,"msg":"处理成功","extend":{"user":{"name":"kasi","age":24,"sex":"m","hobby":null,"address":"中国"}}} \*/