

这里就直接用我正在开发的洛林博客的里面的写法举例

json是一种数据交换格式，把前后端交互看作是一个镖局运送货物，json就是一种装货物的方式，要考虑的就是后端怎么存取，前端怎么存取

前端的话，json是要和ajax一起使用的，一般是一个js方法，里面包含一个ajax请求，请求数据类型就采用json

就用这个举例了，体积小，样样具到

```
function getOneArtical(id) {
    $.ajax("${pageContext.request.contextPath}/getOneArtical", {
        dataType : "json",
        type : "post",
        contentType : "application/json",
        data : JSON.stringify({
            articalId : id,
        }),
        async : true,
        success : function(data) {
            window.location.href = 'readArtical';
        },
        error : function() {
            alert("数据发送失败");
        }
    });
}
```

这个js方法看名字是获取一篇文章，还有个参数id，就是该文章的id，方法里面是一个ajax请求，映射路径就是“/getOneArtical”，dataType预期了服务器返回的数据类型，type指定请求方式，contentType就是指定了传送信息到后端的数据类型，data存放了转换为json格式的数据，大括号里写法是键值，中间“:”隔开，最后“,”收尾，async指定为异步请求，success是成功获取到后端传过来的信息后（可能就是通知一下，我收到你的信息了，也必须通知，不然前端会按照没发送成功处理）做一些处理，这里是跳转页面，error则就是没有收到后端回馈信息时，自动理解为数据传输失败，报错

success后面function的参数data其实就是获取了后端传过来的json格式数据，调用方式就是data.相应键，如果相应值是对象，继续这么...下去，就是自己想要的那个值

我们再来看看后端

```
@RequestMapping(value = "/getOneArtical")
public void getOneArtical(@RequestBody String harvest, HttpServletResponse response,
    HttpSession session) throws Exception {
    JSONObject js = JSONObject.fromObject(harvest);
    Integer id = (Integer) js.get("articalId");
    System.out.println(id);
    Artical artical = articalService.selectArticalById(id);
    session.setAttribute("artical", artical);

    JSONObject json = new JSONObject();
    json.accumulate("success", "success");
    response.setContentType("text/html;charset=UTF-8");
    response.getWriter().append(json.toString());
}
```

@RequestBody注解修饰的String类型的参数其实就是获取了整个一串json数据，然后通过JSONObject.fromObject转换成JSONObject类型对象，然后通过get相应的键获取相应的值 向前端传数据就如代码下半段所示