@ResponseBody与@RequestBody注解

# @ResponseBody注解

## 作用：

**@Responsebody 注解**表示该方法的返回的结果**直接写入 HTTP 响应正文（ResponseBody）中**，并调用适合**HttpMessageConverter**的Adapter转换对象，写入输出流，一般在**异步获取数据时使用**。(通常是在使用 @RequestMapping 后，返回值通常解析为跳转路径，加上 @Responsebody 后返回结果不会被解析为跳转路径，而是直接写入HTTP 响应正文中。)

需要通过适当的**HttpMessageConverter**转换为指定格式后，写入到Response对象的body数据区。

## 使用场合

异步获取数据；返回的数据不是html标签的页面，而是其他某种格式的数据时（如json、xml等）使用；如典型的应用AJAX请求。

## 示例：AJAX请求

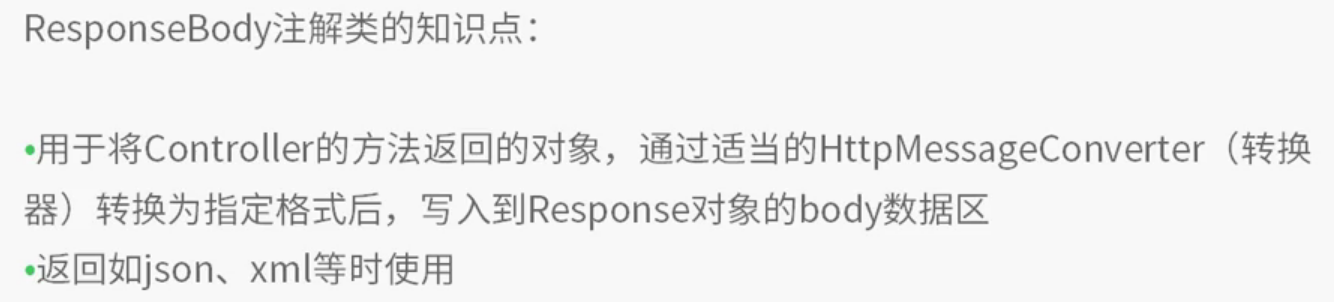
异步获取 json 数据，加上 @Responsebody 注解后，就会直接返回 json 数据。



前端页面的AJAX请求：



## HttpMessageConverter转换器









# @RequestBody

## 功能：

@RequestBody 注解则是将 **HTTP 请求正文**插入方法中，使用适合的 **HttpMessageConverter** 将请求体写入某个对象。

@RequestBody 将HTTP请求正文转换为适合的HttpMessageConverter对象。

## 作用：

### 该注解用于读取Request请求的body部分数据，使用系统默认配置的***HttpMessageConverter***进行解析，然后把相应的数据绑定到要返回的对象上；

### 再把HttpMessageConverter返回的对象数据绑定到 controller中方法的参数上。

## 使用时机：

A) GET、POST方式提时， 根据request header Content-Type的值来判断:

application/x-www-form-urlencoded， 可选（即非必须，因为这种情况的数据@RequestParam, @ModelAttribute也可以处理，当然@RequestBody也能处理）；

multipart/form-data, 不能处理（即使用@RequestBody不能处理这种格式的数据）；

其他格式， 必须（其他格式包括application/json, application/xml等。这些格式的数据，必须使用@RequestBody来处理）；

B) PUT方式提交时， 根据request header Content-Type的值来判断:

application/x-www-form-urlencoded， 必须；multipart/form-data, 不能处理；其他格式， 必须；

说明：request的body部分的数据编码格式由header部分的Content-Type指定；

## @RequestBody示例

@RequestMapping(value = "user/login")

@ResponseBody

// 将ajax（datas）发出的请求写入 User 对象中

public User login(@RequestBody User user) {

// 这样就不会再被解析为跳转路径，而是直接将user对象写入 HTTP 响应正文中

return user;

}

# @ResponseBody与@RequestBody的区别

两者的区别很大：

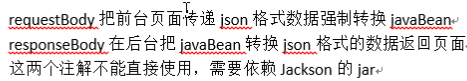
@ResponseBody对应的是返回的数据直接写入body区；不经过视图解析；

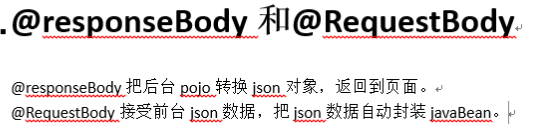
而@RequestBody将**HTTP请求正文**插入方法中,使用适合的**HttpMessageConverter**将请求体写入某个对象。

# Spring MVC对AJAX的支持概括

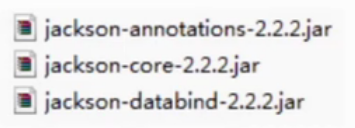
## 两个注解：@RequestBody和@ResponseBody

**Spring MVC**对AJAX的支持通过**@RequestBody和@ResponseBody**这两个注解实现。





## @RequestBody和@ResponseBody注解必须依赖Jackson的jar包



# RequestBody：解析json字符串

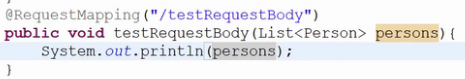
在**Controller**的handle方法中，如何获取AJAX发送的请求参数：

## 如果发送的是一个简单的Json字符串，按照以往的做法，直接在handle方法的入参中加入指定类型的数据，就可以获取参数值。

如：简单的json语句：data:{"name":"zhaohong","age":"25"},

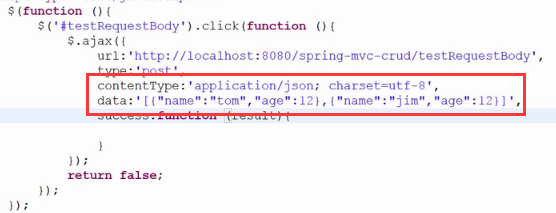
利用Person person就可以接收。





## 但是，当发送一个数组形式的json字符串时，如何接收呢？

对于数组形式的**json字符串**，直接利用**List接收**，报出错误。





意思就是，**默认情况下，Spring MVC不能解析复杂的json字符串**。

## @RequestBody使用位置

@**RequestBody**用于修饰**controller类**的handle方法的形参（入参）。



## @RequestBody的作用

@RequestBody 注解则是将 HTTP 请求正文插入方法中，使用适合的 **HttpMessageConverter** 将请求体写入某个对象。

1) 该注解用于读取Request请求的body部分数据，使用系统默认配置的HttpMessageConverter进行解析，然后把相应的数据绑定到要返回的对象上；

2) 再把**HttpMessageConverter**返回的对象数据绑定到 controller中方法的参数上。

## @RequestBody：主要用来解析AJAX的复杂的请求json字符串

利用**@RequestBody**修饰**handle方法的入参**，解析AJAX的**复杂的json字符串**。



注意：

### AJAX的data部分：用方括号括起来，引号把方括号括起来，如

简单的json： data:{"name":"zhaohong","age":"25"},

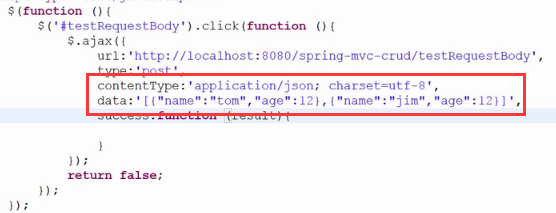
数组形式的json:data:'[{"name":"zhaohong","age":"25"},{"name":"zhaoxiyuan","age":"15"}]',

### AJAX必须制定数据类型contentType

由于Spring MVC只能解析json类型的数据，所以必须指定contentType，否则无法解析。

request的**body部分**的数据编码格式由header部分的**Content-Type**指定；

**contentType:'application/json;charset=utf-8',**



# @ResponseBody

## @ResponseBody使用场景与时机

一般情况下，未经@R**esponseBody修饰的方法，需要经过视图处理器，返回一个页面。但是对于异步加载部分数据，如AJAX请求时，**需要返回的数据不是**html标签的页面**，而是**其他某种格式的数据时**（如json、xml等），这时候就要使用@R**esponseBody修饰**；

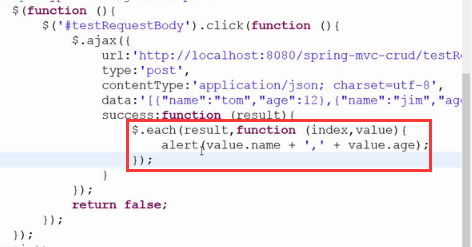
@R**esponseBody一般在异步获取数据时使用**，在使用@RequestMapping后，返回值通常解析为跳转路径，**加上@responsebody后返回结果不会被解析为跳转路径，而是直接写入HTTP response \*body中。也就是说，不再经过视图处理器。**

通常用于**AJAX异步请求数据**：异步获取json数据，加上@responsebody后，会直接返回json数据。

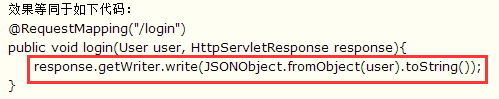
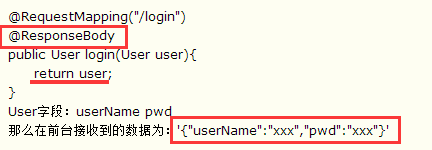
## 示例

### 示例1





### 示例2



### 示例3





