# 需求

Js需要跨域向其他服务器发送请求

通过js可以直接发送Ajax请求到本域的服务端获取数据，但是如果出现跨域，就会复杂些。

# 域

域：同一ip，同一端口

跨域：不同ip，或者相同ip不同端口

# 前端实现

使用**getJSONP()**方法，两个参数

参数1：请求服务器的url，需要一个参数，值为回调方法的方法名，参数名需要和服务端协商。

参数2：回调方法，请求成功后执行的方法，方法参数，返回的json对象。请求成功后会立即调用此方法。

例：

**URL\_Serv**: **"http://127.0.0.1:8081/rest/itemcat/all?callback=category.getDataService"**

**getDataService:**

getDataService: **function**(data) {. . . }

**$**.getJSONP(**this**.**URL\_Serv**, ***category***.getDataService);

# 后端实现

## 实现一，手动拼串

Callback属性为传输过来的方法名，查询数据后转化为json字符串，需要进行一步拼串，拼成js执行代码，返回后会被前端直接执行。

方法名(json数据);

@RequestMapping(value = **"/itemcat/all"**, produces = {MediaType.***APPLICATION\_JSON\_VALUE*** + **";charset=utf-8"**}) *//指定字符编码，防止乱码*  
**public** String getItemCatAll(@RequestParam(**"callback"**) String callback) {

*//获取需要返回的数据*   
 CatResult catList = **itemCatService**.getItemCatList();

*//把对象转化为json字符串*  
 String json = JsonUtils.*objectToJson*(catList);  
 *//拼串，js方法(json串); jsonp直接调用* json = callback + **"("** + json + **")"** + **";"**;  
 **return** json;  
}

## 实现二，框架实现

SpringMVC在4.1版本后，封装这个功能

需要使用**MappingJacksonValue**类的对象

@RequestMapping(value = **"/itemcat/all"**, produces = {MediaType.***APPLICATION\_JSON\_VALUE*** + **";charset=utf-8"**})  
**public** Object getItemCatAll(@RequestParam(**"callback"**) String callback) {  
 *//获取需要返回的数据* CatResult catList = **itemCatService**.getItemCatList();  
 *//创建一个MappingJacksonValue对象，传入需要返回的数据* MappingJacksonValue jacksonValue = **new** MappingJacksonValue(catList);  
 *//设置jsonp调用的方法名* jacksonValue.setJsonpFunction(callback);  
 *//返回这个MappingJacksonValue对象* **return** jacksonValue;  
}