**jquery chosen插件使用及select常用方法**

[JuanitaLee](https://www.jianshu.com/u/03a68162b7dc)关注

0.4822018.11.14 17:09:17字数 348阅读 13,318

**1、chosen插件使用**

chosen插件依赖于jQuery库或prototype，使用之前要先引入jQuery或prototype。

引入jquery插件和chosen插件，对需要美化的下拉框执行chosen函数：$(selector).chosen(options)。可以把selector写成样式类，这样执行完chosen函数后，只要加上样式类就可以了。

**举例**

<head>

<meta charset="utf-8">

<link href='https://cdn.bootcss.com/chosen/1.5.1/chosen.min.css' rel='stylesheet'/>

<style>

.w-country{width:100px;}

</style>

<script src='https://code.jquery.com/jquery-1.9.0.js'></script>

<script src='https://cdn.bootcss.com/chosen/1.5.1/chosen.jquery.js'></script>

<script>

$(function(){

$('.chosen').chosen({

no\_results\_text: "没有找到结果！",//搜索无结果时显示的提示

search\_contains:true, //关键字模糊搜索。设置为true，只要选项包含搜索词就会显示；设置为false，则要求从选项开头开始匹配

allow\_single\_deselect:true, //单选下拉框是否允许取消选择。如果允许，选中选项会有一个x号可以删除选项

disable\_search: false, //禁用搜索。设置为true，则无法搜索选项。

disable\_search\_threshold: 0, //当选项少等于于指定个数时禁用搜索。

inherit\_select\_classes: true, //是否继承原下拉框的样式类，此处设为继承

placeholder\_text\_single: '选择国家', //单选选择框的默认提示信息，当选项为空时会显示。如果原下拉框设置了data-placeholder，会覆盖这里的值。

width: '400px', //设置chosen下拉框的宽度。即使原下拉框本身设置了宽度，也会被width覆盖。

max\_shown\_results: 1000, //下拉框最大显示选项数量

display\_disabled\_options: false,

single\_backstroke\_delete: false, //false表示按两次删除键才能删除选项，true表示按一次删除键即可删除

case\_sensitive\_search: false, //搜索大小写敏感。此处设为不敏感

group\_search: false, //选项组是否可搜。此处搜索不可搜

include\_group\_label\_in\_selected: true //选中选项是否显示选项分组。false不显示，true显示。默认false。

});

$('.chosen2').chosen({

search\_contains:false,

enable\_split\_word\_search: true //分词搜索，选项词可通过空格或'[]'分隔。search\_contains为false时才能看出效果

});

});

</script>

</head>

<body>

<select class='chosen w-country' data-placeholder='请选择国家' multiple >

<option value=""></option>

<option value="China">中国</option>

<option value="US">US</option>

<option value="England">英国</option>

<option value="Canada">加拿大</option>

<option value="Cube">古巴</option>

</select>

<br/><br/>

<select class='chosen'>

<optgroup label="Swedish Cars">

<option value ="volvo">Volvo</option>

<option value ="saab">Saab</option>

</optgroup>

<optgroup label="German Cars">

<option value ="mercedes">Mercedes</option>

<option value ="audi">Audi</option>

</optgroup>

</select>

<br/><br/>

<select class='chosen2' >

<option value=""></option>

<option value="vegetable">西红柿 黄瓜 芹菜</option>

<option value="fruit">葡萄 草莓 芭乐[潘石榴]</option>

</select>

</body>

**属性：**

| **选项** | **描述** |
| --- | --- |
| data-placeholder | 占位符文字。注意： 该属性会覆盖 placeholder\_text\_multiple 或 placeholder\_text\_single 选项。 |
| multiple | 有此属性的 select 会渲染成可以多选的 Chosen 选框 |
| selected, disabled | 设置选中、禁止状态，Chosen 会读取这些属性 |

<select class='chosen w-country' data-placeholder='请选择国家' multiple >

<option value=""></option>

<option value="China" selected>中国

<option value="Canada" disabled>加拿大</option>

</select>

**监听事件：**

| **选项** | **描述** |
| --- | --- |
| change | Chosen 触发标准的 change 事件，同时会传递 selected or deselected 参数， 方便用户获取改变的选项 |
| chosen:ready | Chosen 实例化完成时触发 |
| chosen:maxselected | 超过 max\_selected\_options 设置时触发 |
| chosen:showing\_dropdown | Chosen 下拉选框打开完成时触发 |
| chosen:hiding\_dropdown | Chosen 下拉选框关闭完成时触发 |
| chosen:no\_results | 搜索没有匹配项时触发 |

$('.chosen').on('change', function(e, params) {

do\_something(e, params);

});

**Chosen 触发的事件:**

| **事件** | **描述** |
| --- | --- |
| chosen:updated | 通过 JS 改变 select 元素选项时应该触发此事件，以更新 Chosen 生成的选框 |
| chosen:activate | 相当于 HTML focus 事件 |
| chosen:open | 激活 Chosen 并显示搜索结果 |
| chosen:close | 关闭 Chosen 并隐藏搜索结果 |

$('.chosen').trigger('chosen:updated');

//我在项目中用的是

$('.chosen').trigger('liszt:updated');

**常用操作：**  
1.1、设置chosen下拉框中的选项内容

/\*设置下拉框选项内容\*/

$(selectId).html(change\_options);

//需要执行trigger方法，才会生效

$(selectId).trigger("liszt:updated");

1.2、设置chosen选中某项

//设置选中第几项

$(selectId).get(0).selectedIndex=i;

//需要执行trigger方法，才会生效

$(selectId).trigger("liszt:updated");

//需要执行change()方法，才会触发onChange事件

$(selectId).change();

**2、select使用操作**

**2.1、设置select内容**

清空select：  
$("#ddlregtype").empty();  
设置select选项  
$("#ddlregtype").html();

**2.2、获取select相关内容**

获取select选中的 value:  
$("#ddlregtype").val();  
获取select 选中的 text :  
$("#ddlregtype").find("option:selected").text();  
获取select选中的索引:  
$("#ddlregtype").get(0).selectedIndex;

**2.3、设置select选中某一项**

<div id="div1">

<select id="select1">

<option value="1">律师</option>

<option value="2">监护人</option>

<option value="3">亲友</option>

<option value="4">其他</option>

</select>

</div>

**根据value值**

//根据value值

$("#select1 option[value='4']").attr("selected","selected");

$("#select1 option[value='4']").attr("selected",true);

$("#select1").val("4");

$("#select1").get(0).value = '3';

**根据索引值**

//根据索引值

$("#select1 ").get(0).selectedIndex=index;//index为索引值（从0开始）

$("#select1").get(0).options[i].selected = true; //i为索引值(从0开始)

**根据text值**

//根据text值

$('#test option[text="b"]').attr("selected",true);

$('#test').find('option[text="b"]').attr("selected",true);

**但是以上两种方法在jquery低于1.4.2的版本（含）中有效，在更高版本中无效！**

【高版本Jquery中解决办法：】  
解决一：精确匹配，选择文本与所给字符串完全一样的option。

$('#select1 option').filter(function(){return $(this).text()=="亲友";}).attr("selected",true);

解决二：子串匹配，选择文本包含所给字符串的option。

$("#select1 option:contains('其他')").attr('selected', true);

解决三：自定义函数

$("#btn").click(function(){

var count=$("#select1").get(0).options.length;

for(var i=0;i<count;i++){

if($("#select1").get(0).options[i].text == "亲友")

{

$("#select1").get(0).options[i].selected = true;

break;

}

}

});

**2.4、操作select option选项**

$("#select\_id").append("<option value='value'>text</option>"); //添加一项option

$("#select\_id").prepend("<option value='0'>请选择</option>"); //在前面插入一项option

$("#select\_id option:last").remove(); //删除索引值最大的option

$("#select\_id option[index='0']").remove();//删除索引值为0的option

$("#select\_id option[value='3']").remove(); //删除值为3的option

$("#select\_id option[text='4']").remove(); //删除text值为4的option