1. 特性及兼容性

**特性**

* 支持点击label实现checkbox组选择.
* 头部选项，如全选/ 取消全选 /关闭功能.
* 支持键盘选择.
* 支持5种不同的事件回调函数.
* 以列表方式显示选中项目，并且可以设置最大显示值.（最大显示值已屏蔽，改为全部显示）
* 方便改变位置，渐变速度，滚动容器的高度，链接文字，文本框默认内容等.
* 最小版只有 6.9kb.

**兼容性**

* Firefox 2 – 3.6
* IE 6 – 8
* Chrome Beta/4
* Safari 4
* Opera 10

**所有属性和回调函数大小写敏感，获取多选框的值时用val(),不能用attr(“value”)(该方法只能获取到该多选框中选中的第一个值);**

1. 基本属性

2.1 Header：下拉框头部分（可选值有true（default）、false和自定义文字）

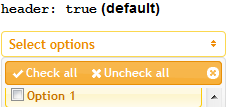


true（default）

$("select").multiselect({

header：true

});

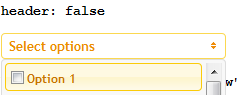


false

$("select").multiselect({

header：false

});

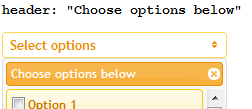


自定义文字

$("select").multiselect({

header：’Choose options below’

});



2.2 height下拉框高度，默认175

$("select").multiselect({

height：175

});

2.3 minWidth下拉框最小宽度，默认225

$("select").multiselect({

minWidth：225

});

2.4 classes下拉框class，默认‘’

$("select").multiselect({

classes：’ className’(要设置的className)

});

2.5 checkAllText下拉框全选文本，可选值有Check all（默认）和自定义文字（可定义中文）

$("select").multiselect({

checkAllText：’Check all’

});

2.6 uncheckAllText下拉框全不选文本，可选值有Uncheck all（默认）和自定义文字（可定义中文）

$("select").multiselect({

uncheckAllText：’Uncheck all’

});

2.7 noneSelectedText下拉框没有选中项时文本框中显示文字，可选值有Select options（默认）和自定义文字（可定义中文）

$("select").multiselect({

noneSelectedText: "==请选择==",

checkAllText: "全选",

uncheckAllText: '全不选',

selectedList:1

});

2.8 selectedText 默认值为‘# selected’，该属性作用为当选中项条数大于 selectedList最多显示选中项条数值时文本框显示的内容，由于selectedList属性作用已经屏蔽，所以该属性已经无效了，用法如下，可直接设置为文本，也可用该回调函数设置

$("select").multiselect({  
   selectedText: function(numChecked, numTotal, checkedItems){  
      return numChecked + ' of ' + numTotal + ' checked';  
   }  
});

2.9 selectedList页面选项框里最多显示选中项条数（默认0条，既只显示’\*selected’），超过后显示 \*selected（以选中\*条选项）(该属性作用已屏蔽但未删除，改成选中多少项就显示多少项)

2.10 show下拉框显示时效果，第一个参数是效果名称（具体可选值不祥，默认是null），第二个参数是延迟\*毫秒，可不设置，默认400ms

2.11 hide 下拉框隐藏时效果，第一个参数是效果名称（具体可选值不祥，默认是null），第二个参数是延迟\*毫秒，可不设置，默认400ms

$("#test-1").multiselect({

   show: ["bounce", 200],

   hide: ["explode", 1000]

});

2.12 autoOpen下拉框在组建加载完成时自动打开，默认是false，可选值只有false和true

$("select").multiselect({

autoOpen：false

});

2.13 multiple选择项是否可多选，可选值只有true（默认）和false

$("select").multiselect({  
   multiple: false,//单选（true为多选）  
   header: "Select an option",  
   noneSelectedText: "Select an Option",  
   selectedList: 1  
});

2.14 position下拉框打开时的位置（需要引入jquery.ui.min.js，jquery需要jquery-1.4.3.js，1.4.2版本有bug）默认{}（在下拉框下方）

$("select").multiselect({  
   position: {  
      my: 'center',  
      at: 'center'  
        
     // only include the "of" property if you want to position  
     // the menu against an element other than the button.  
     // multiselect automatically sets "of" unless you explictly  
     // pass in a value.  
   }  
});

$("select").multiselect({  
   position: {  
      my: 'left bottom',  
      at: 'left top'  
   }  
});

2.15 appendTo 默认值为body

2.16 disabled在<option disabled="disabled">disabled属性可使该选项变为不可选

===以下蓝色部分为自行添加的属性值

2.17 url 请求路径，该属性值不为null或者“”（默认值）的情况下，该组件可根据url进行ajax请求返回的json数据进行自动添加选项

2.18 data 传入的json类型数据，，该属性值不为null或者“”（默认值）的情况下，该组件可直接根据data（var data=[{'text':'飞','value':'9'},{'text':'发','value':'10'},{'text':'放','value':'13'}]）中的json数据进行自动添加选项(若既有参数值url又有参数值data，根据data创建的选项排在根据ajax请求参数创建的选项后面)(创建下拉树时也是往这个参数传json对象，具体格式参照zTreedemo，格式类似于：

var zNodes =[

{ name:"父节点1 - 展开", open:true,

children: [

{ name:"父节点11 - 折叠",

children: [

{ name:"叶子节点111"},

{ name:"叶子节点112"},

{ name:"叶子节点113"},

{ name:"叶子节点114"}

]},

{ name:"父节点12 - 折叠",

children: [

{ name:"叶子节点121"},

{ name:"叶子节点122"},

{ name:"叶子节点123"},

{ name:"叶子节点124"}

]},

{ name:"父节点13 - 没有子节点", isParent:true}

]},

{ name:"父节点2 - 折叠",

children: [

{ name:"父节点21 - 展开", open:true,

children: [

{ name:"叶子节点211"},

{ name:"叶子节点212"},

{ name:"叶子节点213"},

{ name:"叶子节点214"}

]},

{ name:"父节点22 - 折叠",

children: [

{ name:"叶子节点221"},

{ name:"叶子节点222"},

{ name:"叶子节点223"},

{ name:"叶子节点224"}

]},

{ name:"父节点23 - 折叠",

children: [

{ name:"叶子节点231"},

{ name:"叶子节点232"},

{ name:"叶子节点233"},

{ name:"叶子节点234"}

]}

]},

{ name:"父节点3 - 没有子节点", isParent:true}

];

2.19 executeMessage 组件进行ajax请求时在选择文本框中的提示信息，默认为“loading...”

2.20 successMessage 组件完成ajax请求时在选择文本框中的提示信息，1秒后恢复为noneSelectedText值，默认为“”,此时不做提示，选择文本框中显示的是noneSelectedText的值

2.21 errorMessage 组件进行ajax请求发生错误时在选择文本框中的提示信息，默认为“”,此时不做提示，选择文本框中显示的是noneSelectedText的值

2.21 setting zTree配置参数的集合，所有的zTree参数都可以配置到这里面，具体配置方式参照zTree的api，（其中setting.check.enable默认设置为跟multiple的值一致，setting.check.radioType默认设置为”all”， setting.check. chkboxType默认设置为{ "Y": "", "N": "" }，同时设置了setting.callback.onClick和setting.callback.onCheck的回调方法，主要设置了点击或选中时往文本框回填选中的值）

1. 基本方法使用

3.1 basic

需引入以下文件

<link rel="stylesheet" type="text/css" href="multiselectSrc/jquery.multiselect.css" />

<link rel="stylesheet" type="text/css" href="assets/style.css" />

<link rel="stylesheet" type="text/css" href="http://ajax.googleapis.com/ajax/libs/jqueryui/1/themes/ui-lightness/jquery-ui.css" />

<script type="text/javascript" src="ui/jquery-1.4.3.js"></script>

<script type="text/javascript" src="ui/jquery.ui.widget.js"></script>

<script type="text/javascript" src="multiselectSrc/jquery.multiselect.js"></script>

<script type="text/javascript" src="ui/jquery.ui.core.js"></script>这个可以换成jquery.ui.min.js

然后再js里直接调用即可

$(function(){  
   $("select").multiselect();   
});

3.2 Show/hide with Animation显示隐藏效果

$(function(){

$("#test-1").multiselect({  
   show: ["bounce", 200],  
   hide: ["explode", 1000]  
});

$("#test-2").multiselect({  
   show: "bounce",  
   hide: "explode"  
});

});

3.3回调函数

2.3.1 click 选项列（既<option>标签内容）被点击时触发，参数ui既是<option>对象

click: function(event, ui){

//do something

},

3.3.2 beforeopen 下拉框打开前触发

beforeopen: function(){

//do something

},

3.3.3 open 下拉框打开时触发

open: function(){

//do something

},

3.3.4 beforeclose 下拉框关闭前时触发

beforeclose: function(){

//do something

},

3.3.5 close 下拉框关闭时触发

close: function(){

//do something

},

3.3.5 checkAll 下拉选项全选时触发

checkAll: function(){

//do something

},

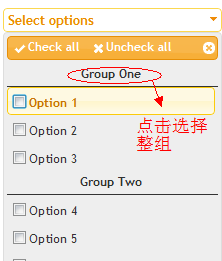
3.3.5 uncheckAll 下拉选项全不选时触发

uncheckAll: function(){

//do something

},

3.3.5 optgrouptoggle 下拉选项组选择时触发（既点击<optgroup >标签内容时，默认点击该标签时可选择整组，如有加入过滤，选中的则是过滤后的整组），参数ui既是<optgroup对象



optgrouptoggle: function(event, ui){

//do something

var values = $.map(ui.inputs, function(checkbox){

return checkbox.value;

}).join(", ");

alert(values);

}

## 3.4 最大选择条数（存在选择组时该函数变成最大可选择2个整组）

var warning = $(".message");  
  
$("select").multiselect({   
   header: "Choose only TWO items!",  
   click: function(e){  
       if( $(this).multiselect("widget").find("input:checked").length > 2 ){  
           warning.addClass("error").removeClass("success").html("You can only check two checkboxes!");  
           return false;  
       } else {  
           warning.addClass("success").removeClass("error").html("Check a few boxes.");  
       }  
   }  
});

## 3.5 设置单选

$("select").multiselect({  
   multiple: false,//单选（true为多选）  
   header: "Select an option",//下拉框头部分提示  
   noneSelectedText: "Select an Option",//未选中任何项时文本框提示  
   //selectedList: 1//已屏蔽，不需要设置  
});

## 3.6 下拉框可用（不可用）

在<select disabled="disabled" >设置该disabled属性的值为’ disabled’后下来框不可用（不可点击，点击无效状态），将属性值改成’enable’后变为可用（可正常使用）

## 3.7 向下拉框中添加选项

$(function(){

var el = $("select").multiselect(),//下拉框对象

disabled = $('#disabled'),//多选框中的一个

selected = $('#selected'),// 多选框中的一个

newItem = $('#newItem');//填写下拉框value和text值得文本框

$("#add").click(function(){//按钮添加新选项事件

var v = newItem.val(), opt = $('<option />', {//选项对象

value: v,

text: v

});

if(disabled.is(':checked')){

opt.attr('disabled','disabled');//默认是否不可选

}

if(selected.is(':checked')){

opt.attr('selected','selected');//默认是否选中

}

opt.appendTo( el );//将选项添加到下拉框中

el.multiselect('refresh');//刷新下拉框内容

});

});

## 3.8 下拉框显示位置

（需要引入jquery.ui.min.js，jquery需要jquery-1.4.3.js，1.4.2版本有bug）默认{}（在文本框下方，当页面位置不够时会自动向下撑开，既向下加大页面长度，下拉框隐藏后页面恢复原来纵向长度）

$("select").multiselect({  
   position: {  
      my: 'center',  
      at: 'center'  
        
     // only include the "of" property if you want to position  
     // the menu against an element other than the button.  
     // multiselect automatically sets "of" unless you explictly  
     // pass in a value.  
   }  
});

$("select").multiselect({  
   position: {  
      my: 'left bottom',  
      at: 'left top'  
   }  
});

## 4 过滤器filter（需要引入jquery.multiselect.filter.css、①jquery.multiselect.filter.min.js（不引入也能实现）和②jquery.multiselect.filter.js文件，以上两个js文件同时引入时一定要按照先①后②的顺序引入，否则没有过滤器出现）

$("select").multiselect().multiselectfilter();

## 4.1 过滤器属性：

## 4.1.1 label过滤器输入框label提示，默认“Filter：”

## The text to appear left of the input. Defaults to "Filter:"

$("select").multiselect().multiselectfilter({

label:’关键字：’

});

## 4.1.2 width过滤器宽度，默认值100px

## The width of the input in pixels. Defaults to 100px in the style sheet, but you can override this for each instance.

$("select").multiselect().multiselectfilter({

width:150

});

## 4.1.3 placeholder过滤器输入框提示文字（既未输入关键字时输入框显示的文字），默认值"Enter keywords"

## The HTML5 placeholder attribute value of the input. Only supported in webkit as of this writing. Defaults to "Enter keywords"

$("select").multiselect().multiselectfilter({

placeholder:’请输入关键字’

});

## 4.1.4 autoReset下拉框关闭时是否自动重置过滤器输入框的值，不会重置选项的选中状态，默认值是false（不重置，既保留关闭前状态），true为重置，既清空过滤器输入框中的值

## A boolean value denoting whether or not to reset the search box & any filtered options when the widget closes. Defaults to false.

$("select").multiselect().multiselectfilter({

autoReset:false

});

## 4.2 事件Events

A callback function that fires after filtering is complete. Accepts two arguments: the original event and an array of matches.

To do something when no matches are found:

$("select").multiselect().multiselectfilter({  
    filter: function(event, matches){   
        if( !matches){  
            // do something  
        }  
    }  
});

To do something with a match:

$("select").multiselect().multiselectfilter({  
    filter: function(event, matches){  
                  
        // find the first matching checkbox  
        var first\_match = $( matches[0] );  
    }  
});

You can also bind to the event after a multiselect has been initialized, like such:

$("select").bind("multiselectfilterfilter", function(event, matches){  
    // do something  
});

### 4.3 函数Methods

Syntax: $("select").multiselectfilter("method\_name");

* **updateCache**

Reloads the cache of values to search against. Make sure you call this after dynamically adding or removing any inputs to the multiselect.

* **destroy**

Destroys the widget.

* **widget**

Returns the wrapper div with the input and label text inside. This is a quick and easy way to access the HTML created by the plugin.