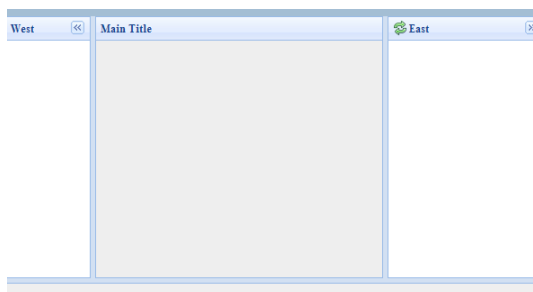


Jquery easyui 使用说明

1. Layout 布局	1
1.1. 样图.....	1
1.2. 示例代码.....	2
2. tabs 面板.....	3
2.1. 样图.....	3
2.2. 示例代码.....	3
3. jQuery EasyUI 提示框(Messenger)用法	5
3.1. 样图.....	5
3.2. 示例代码.....	5
4. 分页(Pagination)用法	6
4.1. 样图.....	7
4.2. 示例代码.....	7
5. jQuery EasyUI 对话框(Dialog)用法	8
5.1. 示例图片.....	8
5.2. 示例代码.....	8
6. jQuery EasyUI 窗口(Window)用法.....	9
6.1. 样图.....	10
6.2. 示例代码.....	10
7. jQuery EasyUI 验证框(ValidateBox)用法	11
7.1. 样图.....	11
7.2. 示例代码.....	11
8. jQuery EasyUI 数字框(NumberBox)用法	13
8.1. 样图.....	13
8.2. 示例代码.....	13
9. DataGrid	13

1. Layout 布局

1.1. 样图



Usage

Markup

在 div 里面加载布局的方法:`class="easyui-layout"`

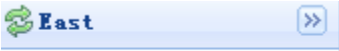

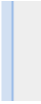
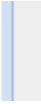

布局面板必需要有一个 'center' 面板.

1.2. 示例代码

```
<div id="cc" style="width:600px;height:400px;" class="easyui-layout">
  <div region="north" title="North Title" split="true" style="height:100px;"></div>
  <div region="south" title="South Title" split="true" style="height:100px;"></div>
  <div region="east" icon="icon-reload" title="East" split="true" style="width:100px;"></div>
  <div region="west" split="true" title="West" style="width:100px;"></div>
  <div region="center" title="center title" style="padding:5px;background:#eee;"></div>
</div>
```

```
jQuery
$('#cc').layout(options);
Dependencies
panel
resizable
Options
Layout Panel Options
```

所有属性都是放在<div/>标签里面

Name	Type	Description	Default
title	string	 标题名字，当写了名字后将会产生  折叠图标	null
region	string	布局的位置值有: north, south, east, west, center.	
border	boolean	显示布局面板的边框	true
split	boolean	 True 时,面板间将产生一个  拖动条可改变面板间大小	false
icon	string	在面板头部显示图标 CSS 将在布局面板中产生一个图标如:  icon="icon-reload"	null
href	string	从远程地址加载数据的 URL	null

fit="true" 自动改变大小

Methods

Name	Parameter	Description
panel	region	Return the specified panel, the 'region' parameter possible values: 'north', 'south', 'east', 'west', 'center'.
collapse	region	Collapse the specified panel, the 'region' parameter possible values: 'north', 'south', 'east', 'west'.
expand	region	Expand the specified panel, the 'region' parameter possible values: 'north', 'south', 'east', 'west'.

2. tabs 面板

2.1. 样图



2.2. 示例代码

在<div/>标签里使用方法 **class="easyui-tabs"**

Usage

Markup

```
<div id="tt" style="width:500px;height:250px;" class="easyui-tabs">
  <div title="Tab1" style="padding:20px;display:none;">
    tab1
  </div>
  <div title="Tab2" closable="true" style="overflow:auto;padding:20px;display:none;">
    tab2
  </div>
  <div title="Tab3" icon="icon-reload" closable="true" style="padding:20px;display:none;">
    tab3
  </div>
</div>
```

jQuery

To create a tabs container

```
$('#tt').tabs(options);
```

To add a tab panel:

```
$('#tt').tabs('add',{
  title:'New Tab',
  content:'Tab Body',
  closable:true
});
```

Dependencies

none

Options

Tabs Container

Override defaults with \$.fn.tabs.defaults.

Properties

Name	Type	Description	Default
width	number	标签容器宽度	auto
height	number	标签容器高度	auto
idSeed	number	应该是生成标签面板的基本 id	0
plain	boolean	标签栏显示背景	false
fit	boolean	设置 true，自适应父集容器大小 false	false
border	boolean	是否显示容器边框	true
scrollIncrement	number	滚动按钮每次被按下时滚动的像素值	100
scrollDuration	number	每次滚动持续的毫秒数	400

Events

Name	Parameters	Description
onLoad	arguments	当一个 AJAX 标签加载远程数据完成时被触发，参数和 jQuery.ajax 成功返回的回调函数相同
onSelect	title	当用户选择一个标签面板时被触发
onClose	title	当用户关闭一个标签面板时被触发

方法名	参数	描述
resize	none	调整标签容器的大小和布局
add	options	添加新标签面板，可选参数是一个配置对象，详细信息可以查看下面的标签面板属性
close	title	关闭一个标签面板，标题参数表明被关闭的面板
select	title	选择一个标签面板
exists	title	指示特定的标签是否存在

方法
标签面板属性

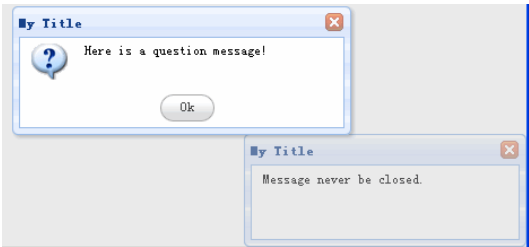
属性名	Type	Description	Default
id	string	标签面板的 ID 属性	null
title	string	标签面板的文本标题	
content	string	标签面板的主体内容	
href	string	填充标签内容的远程 URL 地址	null
cache	boolean	如果为 true，当设置 href 时，对标签面板进行缓存	true
icon	string	标签面板上标题的图标 CSS 类	null
closable	boolean	如果为 true，标签面板会显示出关闭按钮，点击可以关闭选项卡面板。	false

selected	boolean	如果为 true，标签标签面板将被选中	false
width	number	标签面板的宽度	auto
height	number	标签面板的高度	auto

3. jQuery EasyUI 提示框(Messenger)用法

jQuery EasyUI 提示框(Messenger)不仅重写了 window 默认的 alert, confirm 和 prompt，而且还增加一些在页面右下角显示的提示框。

3.1. 样图



jQuery EasyUI 提示框(Messenger)不仅强大，而且也不用任何的 HTML 代码，只需要按照《jQuery EasyUI 框架使用文档》包含必要文件后，在\$(function(){ }); 里插入下面的代码即可：

3.2. 示例代码

```
1. $.messenger.show()  
2. $.messenger.alert('My Title', '事件已经触发');
```

Messenger 需要依存于以下的这些组件：

- * 可拖放(Draggable)
- * 调整大小(Resizable)
- * 面板(panel)
- * 窗口(window)
- * 链接按钮(linkbutton)

上面只是一个小例子，下面来看一下具体的选项和方法：

名字	类型	描述	默认值
ok	字符串	Ok 按钮上的文本	Ok
cancel	字符串	Cancel 按钮上的文本	Cancel

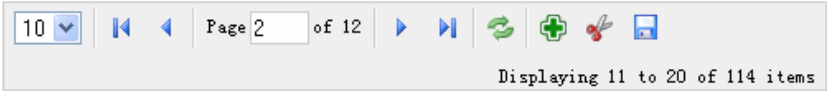
方法

方法名	参数	描述
\$.messenger.show	options	<p>在屏幕的右下角显示一个消息窗口。这些选项的参数可以是一个配置对象：</p> <p>showType: 定义如何将显示消息窗口。可用的值是：<code>null,slide,fade,show</code>。默认值是 <code>slide</code>。</p> <p>showSpeed: 定义消息窗口完成的时间（以毫秒为单位），默认值 <code>600</code>。</p> <p>width: 定义消息窗口的宽度。默认值 <code>250</code>。</p> <p>height: 定义消息窗口的高度。默认值 <code>100</code>。</p> <p>msg: 定义显示的消息文本。</p> <p>title: 定义显示在标题面板显示的标题文本。</p> <p>timeout: 如果定义为 <code>0</code>，消息窗口将不会关闭，除非用户关闭它。如果定义为非 <code>0</code> 值，当超时后消息窗口将自动关闭。</p>
\$.messenger.alert	title, msg, icon, fn	<p>显示一个警告窗口。参数如下：</p> <p>title: 显示在标题面板的标题文本。</p> <p>msg: 提示框显示的消息文本。</p> <p>icon: 提示框显示的图标。可用的值是：<code>error,question,info,warning</code>。</p> <p>fn: 当窗口关闭时触发的回调函数。</p>
\$.messenger.confirm	title, msg, fn	<p>显示一个含有确定和取消按钮的确认消息窗口。参数如下：</p> <p>title: 显示在标题面板的标题文本。</p> <p>msg: 确认消息窗口显示的消息文本。</p> <p>fn(b): 当用户点击按钮后触发的回调函数，如果点击 OK 则给回调函数传 <code>true</code>,如果点击 cancel 则传 <code>false</code>。</p>
\$.messenger.prompt	title, msg, fn	<p>显示一个确定和取消按钮的信息提示窗口，提示用户输入一些文本。参数如下：</p> <p>title: 显示在标题面板的标题文本。</p> <p>msg: 提示窗口显示的消息文本。</p> <p>fn(val): 用户点击按钮后的回调函，参数是用户输入的内容。</p>

4. 分页(Pagination)用法

在 web 开发中，经常会遇到分页的需求，如果靠我们手写代码，可能代码又多，样式又不好看，下面介绍一下 jQuery easyUI 分页功能，非常的方便简单，下面看一下

4.1. 样图



4.2. 示例代码

```
1. <div id="pp" style="background:#efefef;border:1px solid #ccc;"></div>
```

然后按照《jQuery EasyUI 框架使用文档》包含必要文件后，只要在\$(function(){ }); 里插入下面的代码即可：

```
1. $('#pp').pagination(options);
```

下面来介绍 Pagination 的具体用法，首先来看属性：

属性名	类型	描述	默认值
total	数字	当分页建立时设置记录的总数量	1
pageSize	数字	每一页显示的数量	10
pageNumber	数字	当分页建立时，显示的页数	1
pageList	数组	用户可以修改每一页的大小，pageList 属性定义了多少种大小可以改变.	[10,20,30,50]
loading	布尔	定义数据是否正在加载	false
buttons	数组	定义自定义按钮，每个按钮包含两个属性： iconCls: 显示背景图像的 CSS 类 handler: 当一个按钮被点击时的处理函数	null
showPageList	布尔	定义是否显示页面列表	true
showRefresh	布尔	定义是否显示刷新按钮	true
beforePageText	字符串	在输入框组件前显示的标签	Page

afterPageText	字符串	在输入框组件后显示的标签	of {pages}
displayMsg	字符串	显示一个页面的信息。	Displaying {from} to {to} of {total} items

事件

事件名	参数	描述
onSelectPage	pageNumber, pageSize	当用户选择一个新页时触发，回调函数包含两个参数： pageNumber: 新页面的页数 pageSize: 新页面的大小
onBeforeRefresh	pageNumber, pageSize	刷新按钮被点击之前触发，如果返回 false 则取消刷新操作
onRefresh	pageNumber, pageSize	刷新以后触发
onChangePageSize	pageSize	当改变页面大小时触发

5. jQuery EasyUI 对话框(Dialog)用法

jQuery EasyUI 对话框(Dialog)用法与 EasyUI 的窗口用法极为相似，最终的效果图也只与窗口的效果图差几个按钮，下面看一下 EasyUI 的对话框效果图：

5.1. 示例图片



5.2. 示例代码

(很多属性可以和 Window 的用法依样，在标记语言中定义，比如说:icon, title 等等)

```
1. <div id="dd" title="My Dialog" style="width:400px;height:200px;">
```



```
2.     Dialog Content.
3. </div>
```

然后按照《jQuery EasyUI 框架使用文档》包含必要文件后，只要在\$(function(){ }); 里插入下面的代码即可：

```
1. $('#dd').dialog(options);
```

Window 需要依存于以下的三个组件：

- * 可拖放(Draggable)
- * 调整大小(Resizable)
- * 面板(panel)
- * 窗口(window)
- * 链接按钮(linkbutton)

下面来介绍 Window 的具体用法，首先来看**属性**大多数的属性和窗口(Window)的属性是相同的，下面列出一些 Dialog 私有的属性：

属性名	类型	描述	默认值
title	字符串	对话框的标题文本	New Dialog
collapsible	布尔	定义是否显示可折叠按钮	false
minimizable	布尔	定义是否显示最小化按钮	false
maximizable	布尔	定义是否显示最大化按钮	false
resizable	布尔	定义对话框是否可编辑大小	false
toolbar	数组	对话框上的工具条，每个工具条包括：text, iconCls, disabled, handler etc.	null
buttons	数组	对话框底部的按钮，每个按钮包括：text, iconCls, handler etc.	null

事件

Dialog 的事件和窗口(Window)的事件相同，可以参考《jQuery EasyUI 窗口(Window)用法》

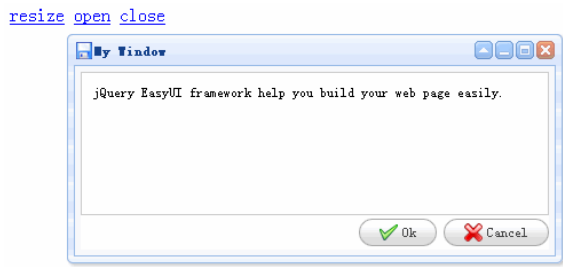
方法

除了” header” 和” body” 以外，Dialog 的函数方法和窗口(Window)的相同，同样也可以参考《jQuery EasyUI 窗口(Window)用法》

6. jQuery EasyUI 窗口(Window)用法

jQuery EasyUI 窗口(Window)的主要用法和面板(panel)用法差不多，首先来看一下 jQuery EasyUI 窗口(Window)在页面里显示的效果是这样的：

6.1. 样图



6.2. 示例代码

(很多 Window 的属性可以在标记语言中定义，比如说:icon, title 等等)

```
1. <div id="win" icon="icon-save" title="My Window">
2.   Window Content
3. </div>
```

然后按照《jQuery EasyUI 框架使用文档》包含必要文件后，只要在\$(function(){ }); 里插入下面的代码即可：
生成一个 Window：

```
1. $('#win').window(options);
```

打开一个 Window：

```
1. $('#win').window('open');
```

- Window 需要依存于以下的三个组件：
- * 可拖放(Draggable)
 - * 调整大小(Resizable)
 - * 面板(panel)

下面来介绍 Window 的具体用法，首先来看**属性**大多数的属性和面板(panel)的属性是相同的，下面列出一些 Window 私有的属性：

属性名	类型	描述	默认值
zIndex	数字	窗口的 z-index 属性，可以通过这个属性来增加	9000
draggable	布尔	定义窗口是否可被拖动	true
resizable	布尔	定义窗口是否可以被改变大小	true
shadow	布尔	如果设置为 true，窗口的阴影也将显示。	true
modal	布尔	定义窗口是否是一个模式窗口。	true

Window 也重写了 Panel 里的一些属性

属性名	类型	描述	默认值
title	字符串	窗口的标题文本	New Window
collapsible	布尔	定义是否显示可折叠定义按钮	true
minimizable	布尔	定义是否显示最小化按钮	true
maximizable	布尔	定义是否显示最大化按钮	true
closable	布尔	定义是否显示关闭按钮	true

事件

Window 的事件和面板 (panel) 的事件相同，可以参考《jQuery EasyUI 面板 (Panel) 用法》

方法

除了” header” 和” body” 以外，Window 的函数方法和面板 (panel) 的相同，同样也可以参考《jQuery EasyUI 面板 (Panel) 用法》

7. jQuery EasyUI 验证框(ValidateBox)用法

jQuery EasyUI 验证框 (ValidateBox) 在表单的验证方面给我们提供了很方便的方法，下面来介绍一下验证框 (ValidateBox) 的详细用法：

7.1. 样图

Name:

Email:

URL:

Note:

This field is required.

7.2. 示例代码

```
1. <input id="vv" required="true" validType="email">
```

然后按照《jQuery EasyUI 框架使用文档》包含必要文件后，只要在\$(function(){ }); 里插入下面的代码即可：

```
1. $('#vv').validatebox(options)
```

下面介绍一下验证规则：

验证规则是通过使用属性 “required” 和” validType” 来定义的，已经实施的规则如下：

- email: 匹配电子邮件正则表达式规则
- url: 匹配 URL 正则表达式规则
- length[0,100]: 允许字符串长度的范围

当然也可以自定义验证规则，重写\$.fn.validatebox.defaults.rules 可以定义一个校验器的功能和无效的显示消息。例如，要定义一个 minLength 有效的类型：

```
1. $.extend($.fn.validatebox.defaults.rules, {
2.     minLength: {
3.         validator: function(value, param){
4.             return value.length >= param[0];
5.         },
6.         message: 'Please enter at least {0} characters.'
7.     }
8. });
```

定义好以后我们就可以使用了，下面的代码表示输入的最小长度是 5 个字符：

```
1. <input class="easyui-validatebox" validType="minLength[5]">
```

下面来介绍更多的用法：**属性**

属性名	类型	描述	默认值
required	布尔	定义文本域是否为必填项	false
validType	字符串	定义字段的验证类型，比如 email, url, etc.	null
missingMessage	字符串	当文本框为空时提示的文本信息	This field is required.
invalidMessage	字符串	当文本框内容不合法时提示的文本信息	null

方法

方法名	参数	描述
destroy	none	删除并且销毁组件
validate	none	做验证以确定文本框的内容是否是有效的。
is Valid	none	调用验证方法并返回验证结果，true 或者 false

8. jQuery EasyUI 数字框(NumberBox)用法

jQuery EasyUI 数字框(NumberBox)相对来说比较简单，就是控制一个输入只接受数字类型，其他所有的字符都不接受。这在我们的应用中还是能起到很大作用的，比如说电话号码框获取金额框是只接受数字的。下面来介绍数字框(NumberBox)用法的详细用法：

8.1. 样图

The Box can only input number.

[disable](#) [enable](#)

8.2. 示例代码

```
1. <input type="text" id="nn"/>
```

然后按照《[jQuery EasyUI 框架使用文档](#)》包含必要文件后，只要在\$(function(){ }); 里插入下面的代码即可：

```
1. $('#nn').numberbox(options);
```

这里的 options 是选项，可以参考下表：

选项名	类型	描述	默认值
min	数字	文本框中可允许的最小值	null
max	数字	文本框中可允许的最大值	null
precision	数字	最高可精确到小数点后几位	0

9. DataGrid

Markup

```
<table id="tt"></table>
```

jQuery

```
$('#tt').datagrid(options);
```

The DataGrid data format sample DataGrid 的数据格式样本

```
{ "total":28,"rows":[{"productid":"FI-SW-01","unitcost":10.00,"status":"P","listprice":16.50,"attr1":"Large","itemid":"EST-1"},
{"productid":"K9-DL-01","unitcost":12.00,"status":"P","listprice":18.50,"attr1":"Spotted Adult Female","itemid":"EST-10"},
```

```
{ "productid": "RP-SN-01", "unitcost": 12.00, "status": "P", "listprice": 18.50, "attr1": "Venomless", "itemid": "EST-11"},
{ "productid": "RP-SN-01", "unitcost": 12.00, "status": "P", "listprice": 18.50, "attr1": "Rattleless", "itemid": "EST-12"},
{ "productid": "RP-LI-02", "unitcost": 12.00, "status": "P", "listprice": 18.50, "attr1": "Green Adult", "itemid": "EST-13"},
{ "productid": "FL-DSH-01", "unitcost": 12.00, "status": "P", "listprice": 58.50, "attr1": "Tailless", "itemid": "EST-14"},
{ "productid": "FL-DSH-01", "unitcost": 12.00, "status": "P", "listprice": 23.50, "attr1": "With tail", "itemid": "EST-15"},
{ "productid": "FL-DLH-02", "unitcost": 12.00, "status": "P", "listprice": 93.50, "attr1": "Adult Female", "itemid": "EST-16"},
{ "productid": "FL-DLH-02", "unitcost": 12.00, "status": "P", "listprice": 93.50, "attr1": "Adult Male", "itemid": "EST-17"},
{ "productid": "AV-CB-01", "unitcost": 92.00, "status": "P", "listprice": 193.50, "attr1": "Adult Male", "itemid": "EST-18"} }
```

Dependencies

resizable

linkbutton

pagination

Options

DataGrid Properties

Override default with \$.fn.datagrid.defaults.

Name	Type	Description	Default
title	string	DataGrid 的面板标题文字。	null
iconCls	string	一个 CSS 类，将提供一个背景图片作为标题图标。	null
border	boolean	真到 DataGrid 中显示面板的边界。	true
width	number	宽度的 DataGrid 的宽度。	auto
height	number	该数据网格的高度的高度。	auto
columns	array	DataGrid 的列的配置对象，看到更多的细节列属性。	null
frozenColumns	array	Columns 属性相同，但将这些列左冻结。	null
striped	boolean	条纹行。	false
method	string	该方法要求远程数据类型。	post
nowrap	boolean	真正显示在同一行数据。	true
idField	string	说明哪些字段是一个标识字段。	null
url	string	一个 URL 请求从远程站点的数据。	null
loadMsg	string	当从远程站点加载数据，显示一个提示信息。	Processing, please wait ... 处理中，请稍候...
pagination	boolean	展现出对 DataGrid 的底部分页工具栏。	false
rownumbers	boolean	显示行号列。	false
singleSelect	boolean	只允许选择一行。	false
fit	boolean	设置大小，以适应它的父容器。	false
pageNumber	number	当设置分页属性，初始化的页码。	1
pageSize	number	当设置分页属性，初始化的页面大小。	10
pageList	array	When set pagination property, initialize the page size selecting list. 当设置分页属性，初始化的页面大小选择清单。	[10,20,30,40,50]
queryParams	object	When request remote data, sending additional parameters also. 当请求，远程数据发送额外的参数也。	{ }

sortName	string	Defines which column can be sorted. 可排序的列定义。	null
sortOrder	string	Defines the column sort order, can only be 'asc' or 'desc'. 定义的列的排序顺序，只能是'递增'或'递减'。	asc

Column Properties

The DataGrid Columns is an array object, which element is an array too. DataGrid 的列是一个数组对象，它是一个数组元素太多。

The element of element array is a config object, which defines every column field. 该元素的数组元素是一个配置对象，它定义了每列字段。

code example:

```
columns:[[
    {field:'itemid',title:'Item ID',rowspan:2,width:80,sortable:true},
    {field:'productid',title:'Product ID',rowspan:2,width:80,sortable:true},
    {title:'Item Details',colspan:4}
],[
    {field:'listprice',title:'List Price',width:80,align:'right',sortable:true},
    {field:'unitcost',title:'Unit Cost',width:80,align:'right',sortable:true},
    {field:'attr1',title:'Attribute',width:100},
    {field:'status',title:'Status',width:60}
]]
```

Name	Type	Description	Default
title	string	The column title text. 列标题文字。	undefined
field	string	The column field name. 列字段的名称。	undefined
width	number	The width of column. 该列的宽度。	undefined
rowspan	number	Indicate how many rows a cell should take up. 显示多少行的单元格应该承担。	undefined
colspan	number	Indicate how many columns a cell should take up. 说明有多少列的单元格应该承担。	undefined
align	string	Indicate how to align the column data. 'left','right','center' can be used. 说明如何对齐列数据。'left','right','center'都可以使用。	undefined
sortable	boolean	True to allow the column can be sorted. 真正让列可以进行排序。	undefined
checkbox	boolean	True to show a checkbox. True 显示一个复选框。	undefined

Events

Name	Parameters	Description
onLoadSuccess	none	Fires when remote data is loaded successfully. 触发时，远程数据加载成功。
onLoadError	none	Fires when some error occur to load remote data. 触发发生时的一些错误加载远程数据。
onClickRow	rowIndex, rowData	Fires when user click a row, the parameters contains: 触发时，用户点击一个行，参数包括： rowIndex: the clicked row index, start with 0 rowData: the record corresponding to the clicked row
onDbClickRow	rowIndex, rowData	Fires when user dblclick a row, the parameters contains: 当用户触发的 DbClick 一排，参数包括：

		<p>rowIndex: the clicked row index, start with 0 rowIndex：单击的行的索引，从 0 开始</p> <p>rowData: the record corresponding to the clicked row rowData：记录点击相应的行</p>
onSortColumn	sort, order	<p>Fires when user sort a column, the parameters contains: 当用户触发排序列, 参数包括:</p> <p>sort: the sort column field name 排序：排序列字段名称</p> <p>order: the sort column order 订购：排序列顺序</p>
onSelect	rowIndex, rowData	<p>Fires when user select a row, the parameters contains: 触发时，用户选择一行，参数包括:</p> <p>rowIndex: the selected row index, start with 0 rowIndex：选定行的索引，从 0 开始</p> <p>rowData: the record corresponding to the selected row rowData：记录对应到选定的行</p>
onUnselect	rowIndex, rowData	<p>Fires when user unselect a row, the parameters contains: 触发时，用户取消选择一个行，参数包括:</p> <p>rowIndex: the unselected row index, start with 0 rowIndex：没有被选中的行的索引，从 0 开始</p> <p>rowData: the record corresponding to the unselected row rowData：记录对应的未选中行</p>

Methods

Name	Parameter	Description
options	none	Return the options object. 返回的选择对象。
resize	none	Do resize and do layout. 做调整，做布局。
reload	none	Reload the rows. 重新加载行。
fixColumnSize	none	fix columns size. 大小固定栏目。
loadData	param	Load local data, the old rows will be removed. 加载本地数据，旧的行会被删除。
getSelected	none	Return the first selected row record or null. 返回第一个选中行的记录或空。
getSelections	none	Return all selected rows, when no record selected, am empty array will return. 返回所有选定行，选择时没有记录，我将返回空数组。
clearSelections	none	Clear all selections. 清除所有选择。
selectRow	index	Select a row, the row index start with 0. 选择一个行，行索引以 0 开始。
selectRecord	id Value	Select a row by passing id value parameter. 选择一行由 ID 值传递参数。
unselectRow	index	Unselect a row. 取消选择 1 行。