## DataGrid

### 使用HTML创建

<table class="easyui-datagrid">

<thead>

<tr>

<th data-options="field:'code'">Code</th>

<th data-options="field:'name'">Name</th>

<th data-options="field:'price'">Price</th>

</tr>

</thead>

<tbody>

<tr>

<td>001</td><td>name1</td><td>2323</td>

</tr>

<tr>

<td>002</td><td>name2</td><td>4612</td>

</tr>

</tbody>

</table>

### 使用JS创建

html:

<table id="dg"></table>

JS:

$('#dg').datagrid({

url: 'datagrid\_data.json',

columns: [[

{ field: 'code', title: 'Code', width: 100 },

{ field: 'name', title: 'Name', width: 100 },

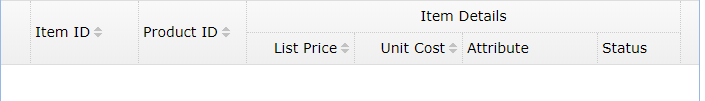
{ field: 'price', title: 'Price', width: 100, align: 'right' }

]]

});

### 常用属性：

columns：数组对象，定义所有列属性。



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 }

]]

设置列属性时的子属性：

title：字符串，定义列标题

field：字符串，定义字段名称，用于绑定数据。

url：字符串，指定数据获取路径。

data：数组或对象，指定加载的数据。数据格式与Json格式相同。

$('#dg').datagrid({

data: [

{ f1: 'value11', f2: 'value12' },

{ f1: 'value21', f2: 'value22' }

]

});

queryParams：对象，当向远端请求数据时，发送的查询参数。

$('#dg').datagrid({

queryParams: {

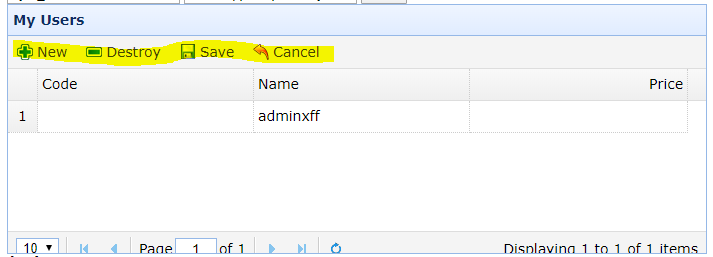
name: 'easyui',

subject: 'datagrid'

}

});

toolbar：顶部工具栏。两个可用的值：数组，显示为按钮；下拉菜单。



A.在toolbar中指明<div>标签名

$('#dg').datagrid({

toolbar: '#tb'

});

<div id="tb">

<a href="#" class="easyui-linkbutton" data-options="iconCls:'icon-edit',plain:true"/a>

<a href="#" class="easyui-linkbutton" data-options="iconCls:'icon-help',plain:true"/a>

</div>

B.在JS中构建

$('#dg').datagrid({

toolbar: [{

iconCls: 'icon-edit',

handler: function () { alert('edit') }

}, '-', {

iconCls: 'icon-help',

handler: function () { alert('help') }

}]

});

### 常用方法：

load：加载并显示第一页的数据行。当指定参数时，将替换原来的数据。通常通过传递一些参数进行查询，可以调用此方法从服务器加载新数据。

$('#dg').datagrid('load', {

code: '01',

name: 'name01'

});

reload：在当前页中刷新数据。

loadData：加载本地数据，旧行将被删除。参数为新的数据

getData：返回加载的数据。

updateRow：更新指定行的数据。有两个参数：index，行索引；row：新的数据

$('#dg').datagrid('updateRow', {

index: 2,

row: {

name: 'new name',

note: 'new note message'

}

});

appendRow：在数据最后插入新一行。

$('#dg').datagrid('appendRow', {

name: 'new name',

age: 30,

note: 'some messages'

});

insertRow：在指定行前插入新一行。有两个参数：index，指定行索引；row，新数据。

// insert a new row at second row position

$('#dg').datagrid('insertRow', {

index: 1, // index start with 0

row: {

name: 'new name',

age: 30,

note: 'some messages'

}

});

deleteRow：删除一行数据，参数为行索引。

getChanges：获取自从上一次提交后修改过的行集合。type参数指示更改行的类型，可能的值为：insert，deleted，updated等。如果未分配type参数，则返回所有更改的行。

acceptChanges：提交自加载以来或上次调用acceptChanges以来的所有更改数据。

rejectChanges：回滚所有变化的数据至自数据首次加载时或到上一次提交时。

## END