

EasyDataTable

AJAX 分页插件使用手册

目录

1	在页面引入 EasyDataTable 核心的 js 和 css 文件	2
2	对 Ajax 分页表格进行数据初始化.....	3
	方法一：给表格添加 easydatatable 类样式即可，等价于方法二的 DataTable.load('tableid')。	3
	方法二：使用 <i>DataTable.load(表格 id, 参数信息)</i> 方法对要进行 Ajax 分页的表格进行数据初始化	3
3	分页表格结构.....	5
4	EasyDataTable Expression Language——表达式语言使用.....	7
4.1	EasyDataTable Property Expression 属性表达式 {数据属性}.....	7
4.2	EasyDataTable Statement Expression 语句表达式 %{表达式语句}%	7
5	EasyTableData 内置数据属性.....	8
6	排序支持.....	9
7	服务器端数据要求.....	10
8	刷新指定数据表格.....	11
9	自定义分页.....	11
9.1	在使用 JavaScript 初始化数据表格时，可指定使用的分页主题：	11
9.2	在分页 DIV 标签上指定分页主题：	11
9.3	取消分页和主题：	11
9.4	自定义分页：	12
10	表格 AJAX 分页实例	12
10.1	指定分页主题 (DataTable.SIMPLE_PAGE), 加入 Loading 提示	12

10.2	判断语句 DataTable 表达式使用，默认 DataTable.SIMPLE_FULL 分页主题 带复选框和自动编号	14
10.3	带搜索条件分页	15
方法一：	给搜索按钮添加 onclick="DataTable.load('当前数据表格 id') "	16
方法二：	给搜索按钮直接加 easydatatable_search 类样式。	16
11	EasyDataTable 分页标签国际化支持	18

使用 Ajax 分页可以提高数据加载和显示速度，减少网络流量，提升客户体验度；同时能够只刷新局部，解决当页面上有多个数据显示表格区域时，传统的分页方式会导致页面全部刷新。

EasyDataTable AJAX 分页插件是基于 jQuery 最好的纯 Ajax 分页插件，简单、易用、灵活；自带分页标签；支持排序；内置 EasyDataTable 表达式语言能够通过 JavaScript 编程增强分页。

快速开发步骤：

1 在页面引入 EasyDataTable 核心的 js 和 css 文件

```
<link rel= "stylesheet" href= "css/datatables.css" type= "text/css"> </link>
<script type= "text/javascript" src= "js/jquery-1.10.2.min.js"> </script>
<script type= "text/javascript" src= "js/easy.datatables.min.js"> </script>
```

2 对 Ajax 分页表格进行数据初始化

方法一：给表格添加 `easydatatable` 类样式即可，等价于方法二的 `DataTable.load('tableid')`。

```
<table class="datatable easydatatable" id="datatable3" width="760px" align="center">
```

方法二：使用 `DataTable.load(表格 id, 参数信息)` 方法对要进行 Ajax 分页的表格进行数据初始化

`DataTable.load(tableid [, easydataParameters]);`

参数说明：

`tableid`：必须参数，显示数据的表格 id

`easydataParameters`：可选参数，指定 EasyDataTable 的参数信息，支撑参数目前支持的参数：

```
{  
    pagetheme: '分页主题',  
    loading: '是否显示加载提示',  
    language: '分页标签的语言'  
}
```

`pagetheme`：可选参数，分页标签主题，支持两套可选分页主题：

`DataTable.FULL_PAGE`（完全主题，默认显示所有分页选项）

每页显示 5 条，共 54 条 共 11 页 首页 上一页 1 2 3 4 5 6 7 下一页 末页 1 >>

`DataTable.SIMPLE_PAGE`（简单主题，不显示当前页前后页快速跳转标签）

每页显示 5 条，共 54 条 共 11 页 首页 上一页 下一页 末页 1 >>

默认值 `DataTable.FULL_PAGE` .

loading: 可选参数, 加载数据时是否显示 loading 提示"[数据正在读取中.....](#)", 可选值为 `true` 或 `false`, 默认为 **false**, 不显示

language: 可选参数, 设置分页标签显示的语言, 默认值为

```
{  
    "first": '首页',  
    "previous": '上一页',  
    "next": '下一页',  
    "last": '末页',  
    "totalCount": '共{0}条',  
    "totalPage": '共{0}页',  
    "rowPerPage": '每页显示{0}条'  
}
```

可根据需要重新定义, {0} 为显示相应数据的占位符, 必须存在

```
<script type="text/javascript">  
$(function() {  
    var pageLanguage={  
        "first": 'first',  
        "previous": 'previous',  
        "next": 'next',  
        "last": 'last',  
        "totalPage": 'total {0} pages',  
        "totalCount": 'total {0} rows',  
        "rowPerPage": 'page for {0} rows',  
    };  
  
    DataTable.load("datatable", {  
        "pagetheme": DataTable.SIMPLE_PAGE,  
        "loading": true,  
        "language": pageLanguage  
    });  
  
    DataTable.load("datatable2");  
    DataTable.load("datatable3");  
    DataTable.load("datatable4");  
});  
</script>
```

3 分页表格结构

```
<form action="服务器分页地址">

<!-- 数据展示表格 -->
<table id="表格 id">
  <!-- 表头行 -->
  <tr><th></th>      ..... </tr>
  <!-- 数据展示行 -->
  <tr><td></td>      ..... </tr>
</table>

<!-- 分页行 -->
<div class="panelBar" style="width: 760px;" size="5, 10, 30, 50">
  </div>
</form>
```

示例：

<!-- 创建表单 action值为分页处理的请求地址 -->

```
<form action="doPage.jsp" name="myform">
```

```
  <div style="height: 260px;overflow:auto;width: 780px;">
```

<!--显示数据的表格，必须指定id，EasyDatatable通过id初始化分页数据-->

```
    <table class="datatable" id="datatable" width="760px"
align="center">
```

<!--表头行-->

```
    <tr>
    <th width="40">
```

<!-- 调用DataTable.checkAll(this,'复选框名称')，可实现复选框

全选/全不选功能 -->

```
    <input type="checkbox" onclick="DataTable.checkAll(this,'mychk')" />
    </th>
```

```
    <th width="80">数量</th>
```

```
    <th width="100">编号</th>
```

```
    <th width="100">姓名</th>
```

```
    <th width="100">信息</th>
```

```
    <th >操作</th>
```

```
  </tr>
```

<!-- 数据展示行，使用 {属性} 属性表达式获取数据 -->

```
  <tr>
    <td style="text-align:center;height: 45px;">
```

<!--自定义复选框,value为对象的id-->

```
    <input type="checkbox" name="mychk" value="{id }"/>
  </td>
```

<!--使用内置的datatableCount属性，显示数据条数-->

```
    <td align="center" style="text-align:center;height:
45px;">{datatableCount }</td>
```

```
    <td style="text-align:center;color:#00f">No.{id}</td>
```

```
    <td align="center">{name}</td>
```

```
    <td>{info}</td>
```

```
    <td align="center" style="width: 120px">
```

```
      <a href="doUser.jsp?o=show&id={id }">查看</a>
```

```

        <a href="doUser.jsp?o=edit&id={id }">修改</a>

        <a href="doUser.jsp?o=delete&id={id }">删除</a>
    </td>
</tr>
</table>
</div>
<!--分页标签部分，使用size属性设置每页显示条数下拉菜单可选值，使用,分隔-->
    <div class="panelBar" style="width: 760px;" size="5,10,30,50">

    </div>
</form>

```

4 EasyDataTable Expression Language——表达式语言使用

4.1 EasyDataTable Property Expression 属性表达式 {数据属性}

属性表达式用于在数据展示行，显示指定的属性值。

在属性表达式中可以直接引用数据属性来获得指定属性的数据，并支持各种数学、比较等 JavaScript 基本运算符进行运算。

{id} {name}

4.2 EasyDataTable Statement Expression 语句表达式 %{表达式语句}%

语句表达式用于在数据展示行，使用编程语句进行控制编程，在语句表达式中支持使用 JavaScript 编写表达式代码；支持直接调用数据属性；也支持使用 EasyDataTable 属性表达式。

语句表达式执行的结果必须使用 EasyDataTable 语句表达式标准输出方法输出：

DataTable.out("内容");

```
<!-- 支持JavaScript语言编写表达式，支持使用 属性名（以变量方式引用和处理） 或
EasyDataTable 属性表达式（必须使用引号定义在字符串中） 直接引用属性值 --%>
%{
    var res=name+" {name}";
    DataTable.out(res);
}%

%{
    if(id%2==0){
        var op='<a href="doUser.jsp?o=show&id='+id+'">查看</a>&nbsp;&nbsp;&nbsp;';
        op+='<a href="doUser.jsp?o=edit&id={id }">修改</a>';
        DataTable.out(op);
    }else{
        DataTable.out('<a href="doUser.jsp?o=show&id={id }">查看
</a>&nbsp;&nbsp;&nbsp;<a href="doUser.jsp?o=edit&id={id }">修改
</a>&nbsp;&nbsp;&nbsp;<a href="doUser.jsp?o=delete&id={id }">删除</a>');
    }
}%
```

5 EasyTableData 内置数据属性

仅在数据展示行有效：

datatableIndex:可获得数据在当前页的索引

datatableCount:可获得数据在当前页的个数

key:Map 数据集合时用来获取数据对于的键

数据展示行+分页标签均可使用：

pageNo:当前页

maxPage:总页数

rowPerPage:每页显示条数

totalCount:数据总条数

order:排序字段

sort:排序方式，desc 或 asc

例如:

当前数据在所有数据中的索引: $\{datatableIndex+(pageNo-1)*rowPerPage\}$

当前数据在所有数据中的个数: $\{datatableCount+(pageNo-1)*rowPerPage\}$

6 排序支持

在**表头行**需要排序字段对应的单元格上添加 **sort=" 排序字段名称"** 属性即可。

```
<tr>
    <th width="80">count</th>
    <th width="80">index</th>
    <th width="80">{index+1}</th>
    <th width="100" order="id">id</th>
    <th width="100" order="name">name</th>
    <th width="100" order="info">info</th>
    <th >操作</th>
</tr>
```

id ↑↓	name ↑↓	info ↑↓	操作
-------	---------	---------	----

可排序列会显示排序箭头。点击即可发送排序字段 order 和排序方式 sort , 实现升序降序切换。

id ↑	name ↑↓	info ↑↓	操作
------	---------	---------	----

服务器端通过 order 和 sort 参数名获取排序信息。

```
String sort = request.getParameter("sort");
```

```
String order = request.getParameter("order");
```

使用 `DataTable.reload('tableid');`可以刷新指定数据表格 , 恢复到没有排序状态。

```
<div onclick="DataTable.reload('datatable3')">刷新</div>
```

7 服务器端数据要求

服务器端必须输出四条如下名称的分页参数 JSON 结果：

data:数据集合，支持 List 和 Map 集合（内置属性 key 可获取 Map 的键，Map 的值无需使用 value.前缀）

pageNo:当前第几页，数字

rowPerPage:每页显示条数，数字

totalCount:数据总条数，数字

可选参数：

[order]:排序字段

[sort]:排序方式，desc 或 asc

如果服务器端分页参数封装在 PageBean 中，例如：

```
public class PageBean {  
    private List data;  
    private int pageNo;  
    private int rowPerPage;  
    private int totalCount;  
  
    //setters&getters .....  
}
```

返回的是 PageBean 对象，则在 table 上加入 value 属性，指定服务器端输出的包含分页参数的分页对象的 json 对象名称：

```
<table class="datatable easydatatable" id="datatable3" width="760px"  
align="center" value="pb">
```

8 刷新指定数据表格

`DataTable.reload("tableId");` //取消排序效果，刷新表格，重新加载数据

9 自定义分页

DataTable 内置分页实现，并提供了两套主题：

`DataTable.FULL_PAGE`（完全主题，默认显示所有分页选项）

每页显示 5 条，共 54 条 共 11 页 首页 上一页 1 2 3 4 5 6 7 下一页 末页 1 >>

`DataTable.SIMPLE_PAGE`（简单主题，不显示当前页前后页快速跳转标签）

每页显示 5 条，共 54 条 共 11 页 首页 上一页 下一页 末页 1 >>

9.1 在使用 JavaScript 初始化数据表格时，可指定使用的分页主题：

```
DataTable.load("datatable",{
    "pagetheme":DataTable.SIMPLE_PAGE
});
```

9.2 在分页 DIV 标签上指定分页主题：

```
<div class="panelBar" style="width: 760px;height: 40px;" size="5,10,30,50"
pagetheme="FULL">
```

```
<div class="panelBar" style="width: 760px;height: 40px;" size="5,10,30,50"
pagetheme="SIMPLE">
```

9.3 取消分页和主题：

`pagetheme="no"`,使用 `display:none` 隐藏，或直接删除分页标签部分即可。

```
<div class="panelBar" style="width: 760px;height: 40px;display: none;"  
size="5,10,30,50" pagetheme="no">
```

9.4 自定义分页：

调用 **DataTable.go(‘加载数据的表格 id’,页数,[每页显示条数])** 函数，即可实现自定义分页跳转。

也可使用

```
<div class="panelBar" style="width: 760px;height: 40px;" size="5,10,30,50"  
pagetheme="no">
```

```
<input type="hidden" name="rowPerPage" value="8"/>
```

当前第{pageNo}页/共{maxPage}页 每页{rowPerPage}条/共{totalCount}条

```
<input type="button" value="首页" onclick="DataTable.go('datatable7',1)"/>
```

```
<input type="button" value="上一页"  
onclick="DataTable.go('datatable7',{pageNo-1})"/>
```

```
<input type="button" value="下一页"  
onclick="DataTable.go('datatable7',{pageNo+1})"/>
```

```
<input type="button" value="末页"  
onclick="DataTable.go('datatable7',{maxPage})"/>  
</div>
```

10 表格 AJAX 分页实例

10.1 指定分页主题 (DataTable.SIMPLE_PAGE) ， 加入 Loading 提示

```
<script type="text/javascript">  
$(function() {  
    DataTable.load("datatable", DataTable.SIMPLE_PAGE, true);  
});
```

```

});
</script>

<div style="margin: 20 0 10 0; font-size: 28px;">
DataTable.SIMPLE_PAGE 分页主题 带Loading提示
</div>

<form action="doPage.jsp" name="myform">
  <div style="height: 260px;overflow:auto;width: 780px;">
    <table class="datatable" id="datatable" width="760px"
align="center">
      <tr>
        <th width="40">
          <input type="checkbox"
onclick="DataTable.checkAll(this,'mychk')" /> <!-- CheckAll -->
        </th>
        <!-- datatableCount -->
        <th width="80">count</th>
        <th width="100">id</th>
        <th width="100">name</th>
        <th width="100">info</th>

        <th >操作</th>
      </tr>

      <!-- 数据展示行 -->
      <tr>
        <td style="text-align:center;height: 45px;">
          <input type="checkbox" name="mychk" value="{id }"/>
        </td>
        <td align="center" style="text-align:center;height:
45px;">{datatableCount }</td>
        <td style="text-align:center;color:#00f">No.{id}</td>
        <td align="center">{name}</td>
        <td>{info}</td>
        <td align="center" style="width: 120px">

          <a href="doUser.jsp?o=show&id={id }">查看</a>

          <a href="doUser.jsp?o=edit&id={id }">修改</a>

          <a href="doUser.jsp?o=delete&id={id }">删除</a>

        </td>
      </tr>
    </table>
  </div>
  <div class="panelBar" style="width: 760px;" size="5,10,30,50">

```

```
</div>  
</form>
```

10.2 判断语句 DataTable 表达式使用 默认 DataTable.SIMPLE_FULL 分页主题 带复选框和自动编号

```
<div style="margin: 40 0 10 0; font-size: 28px;">判断语句DataTable表达式  
使用</div>  
  
<form action="doPage.jsp" name="myform">  
  <div style="height: 260px;overflow:auto;width: 780px;">  
    <table class="datatable easydatatable" id="datatable3"  
width="760px" align="center">  
      <tr>  
        <!-- checkbox -->  
        <th width="40">  
          <input type="checkbox"  
onclick="DataTable.checkAll(this,'mychk')" /> <!-- CheckAll -->  
        </th>  
        <!-- datatableIndex,datatableCount -->  
        <th width="80">count</th>  
        <th width="80">index</th>  
        <th width="80">{index+1}</th>  
        <th width="100">id</th>  
        <th width="100">name</th>  
        <th width="100">info</th>  
  
        <th >操作</th>  
      </tr>  
  
      <!-- 数据展示行 -->  
  
      <tr>  
        <td style="text-align:center;height: 45px;">  
          <input type="checkbox" name="mychk" value="{id }"/>  
        </td>  
        <td align="center">  
{datatableCount+ (pageNo-1) *rowPerPage}</td>  
        <td align="center">  
{datatableIndex+ (pageNo-1) *rowPerPage}</td>  
        <td  
align="center">{datatableIndex+ (pageNo-1) *rowPerPage+1}</td>
```

```
<td style="text-align:center;color:#00f">No.{id}</td>  
<td align="center">{name}</td>  
<td>{info}</td>  
<td align="center">  
  
    <!-- DataTable 表达式 -->  
  
    %{  
        if(id%2==0){  
            var op='<a href="doUser.jsp?o=show&id='+id+'">查看  
</a>&nbsp;&nbsp;&nbsp;&';  
            op+='<a href="doUser.jsp?o=edit&id={id }">修改</a>';  
            DataTable.out(op);  
        }else{  
            DataTable.out('<a href="doUser.jsp?o=show&id={id }">  
查看</a>&nbsp;&nbsp;&nbsp;&<a href="doUser.jsp?o=edit&id={id }">修改  
</a>&nbsp;&nbsp;&nbsp;&<a href="doUser.jsp?o=delete&id={id }">删除</a>');  
        }  
    } %  
  
    </td>  
</tr>  
</table>  
</div>  
    <div class="panelBar" style="width: 760px;" size="5,10,30,50">  
  
    </div>  
  
</form>
```

10.3 带搜索条件分页

搜索按钮数据提交方法：

方法一：给搜索按钮添加 `onclick="DataTable.load('当前数据表格 id')"`

```
<input type="button" class="btn_test"
onclick="DataTable.load('datatable4')" value="立即查询"/>
```

方法二：给搜索按钮直接加 `easydatatable_search` 类样式。

```
<input type="button" class="btn_test easydatatable_search" value="立即查询" />
```

```
<div style="margin: 40 0 10 0; font-size: 28px;">带搜索条件分页</div>
<form action="doPage.jsp" name="myform">
<!--搜索条件部分，通过搜索按钮查询-->
<div style="margin: 20px 0px;">
    用户名: <input type="text" name="user.name" class="txt_test"/>
    信息: <input type="text" name="user.info" class="txt_test"/>
    <input type="button" class="btn_test"
onclick="DataTable.load('datatable4')" value="立即查询"/>
</div>

<div style="height: 260px;overflow:auto;width: 780px;">

    <table class="datatable easydatatable" id="datatable4"
width="760px" align="center">
        <tr>
            <!-- checkbox -->
            <th width="40">
                <input type="checkbox"
onclick="DataTable.checkAll(this,'mychk')" /> <!-- CheckAll -->
            </th>
            <!-- datatableIndex,datatableCount -->
            <th width="80">count</th>
            <th width="80">index</th>
```



```

        <th width="80">{index+1}</th>
        <th width="100">id</th>
        <th width="100">name</th>
        <th width="100">info</th>

        <th >操作</th>
    </tr>

    <!-- 数据展示行 -->

    <tr>
        <td style="text-align:center;height: 45px;">
            <input type="checkbox" name="mychk" value="{id }"/>
        </td>
        <td
align="center">{datatableCount+(pageNo-1)*rowPerPage}</td>
        <td
align="center">{datatableIndex+(pageNo-1)*rowPerPage}</td>
        <td
align="center">{datatableIndex+(pageNo-1)*rowPerPage+1}</td>
        <td style="text-align:center;color:#00f">No.{id}</td>
        <td align="center">{name}</td>
        <td>{info}</td>
        <td align="center">

            <!-- DataTable 表达式 -->

            %{
                if(id%2==0){
                    DataTable.out('<a href="doUser.jsp?o=show&id={id }"
target="ajax">查看</a>&nbsp;&nbsp;&nbsp;<a href="doUser.jsp?o=edit&id={id }"
target="ajax">修改</a>');
                }else{
                    DataTable.out('<a href="doUser.jsp?o=show&id={id }"
target="ajax">查看</a>&nbsp;&nbsp;&nbsp;<a href="doUser.jsp?o=edit&id={id }"
target="ajax">修改</a>&nbsp;&nbsp;&nbsp;<a href="doUser.jsp?o=delete&id={id }"
target="ajax">删除</a>');
                }
            }%

        </td>
    </tr>
</table>
</div>

```

```
<div class="panelBar" style="width: 760px;" size="5,10,30,50">

</div>

</form>
```

11 EasyDataTable 分页标签国际化支持

EasyDataTable 自带了分页标签，需要自定义显示的文字和语言时，标签中的文字可通过 language 参数调整和修改。

- 默认分页标签文字和语言：

```
{
    "first": '首页',
    "previous": '上一页',
    "next": '下一页',
    "last": '末页',
    "totalCount": '共{0}条',
    "totalPage": '共{0}页',
    "rowPerPage": '每页显示{0}条'
}
```

每页显示 5 条，共 54 条 共 11 页 首页 上一页 1 2 3 4 5 6 7 下一页 末页 1 >>

- 自定义分页标签文字和语言：

```
var pageLanguage={
    "first": 'first',
    "previous": 'previous',
    "next": 'next',
    "last": 'last',
    "totalPage": 'total {0} pages',
    "totalCount": 'total {0} rows',
    "rowPerPage": 'page for {0} rows',
};
```

```
DataTable.load("datatable",{  
    "pagetheme":DataTable.SIMPLE_PAGE,  
    "loading":true,  
    "language":pageLanguage  
});
```

page for 5 rows , total 54 rows total 11 pages first previous next last 1 >>

默认分页配置定义在 DataTable 对象的 MSG 属性中 ,可通过修改和重新定义 , 配置为默认分页文字和语言。