



B-JUI 中文手册

本文是 B-JUI 中文使用手册，包括使用示例代码，感兴趣的同学参考下。

概览

B-JUI 仅有一个主页面(document)，框架内的所有子页面将通过 Ajax 获取后作为一个页面片段附加到主页面上，外部页面则通过 iframe 嵌入主页面， 本节介绍 B-JUI 的主页面结构。

HTML5 文档类型

同 Bootstrap， B-JUI 使用 HTML5 文档类型，参照下面的格式进行设置。

```
<!DOCTYPE html>
<html lang="zh-CN">
...
</html>
```

主页面结构(仅 body 部分)

主页面由上（页头）、中左（导航菜单）、中右（工作区）、下（页脚）四部分组成，其中左侧导航菜单可收缩。结构如下：

```
<header class="bjui-header" id="bjui-header">
  <!-- 页头 -->
</header>
<div class="bjui-leftside" id="bjui-leftside">
  <!-- 导航菜单 -->
</div>
<div id="bjui-container">
  <!-- 工作区 -->
</div>
<footer class="bjui-footer" id="bjui-footer">
  <!-- 页脚 -->
</footer>
```

子页面（即页面片段[后面简称：页片]）结构

页片通常由两部分组成，也可以只保留 bjui-pageContent 部分，其中 bjui-pageContent 部分可选 bjui-headBar（顶部工具条）和 bjui-footBar（底部工具条）。结构如下：

```
<div class="bjui-pageHeader">
  <!-- 顶部模块[如：功能按钮、搜索面板] -->
</div>
<div class="bjui-pageContent">
  <div class="bjui-headBar">
    <!-- 顶部工具条 -->
  </div>
  <div data-layout-h="0">
```

```

        <!-- 内容区 -->
    </div>
    <div class="bjui-footBar">
        <!-- 底部工具条 -->
    </div>
</div>

```

`data-layout-h` 属性表示该容器为页片自适应布局，当值为 0 时，B-JUI 会为该容器的高度自动赋值为：本页片总高度减去本页片中的固定元素(`bjui-pageHeader`\`bjui-headBar`\`bjui-footBar`)高度。

`data-layout-h` 属性值不等于 0 时，该容器高度为本页片总高度减去属性值。

需要自定义固定元素(块级元素有效)，请为该元素添加属性 `data-layout-fixed="true"`

完整的页片详见 B-JUI 源代码的 `table.html`，仅 `bjui-pageContent` 部分的页片详见 `form.html`

元素 ID 命名规范

因为本框架默认所有内容都位于一个 Document 中，所以为元素命名 ID 的时候需要做到唯一性，如果确实不可避免的会出现有重复 ID 的现象，需要操作当前页片的元素时，尽量用：

`$.CurrentNavtab.find('#dom-id')`，在当前标签工作区中查找指定 ID 的元素。

或 `$.CurrentDialog.find('#dom-id')`，在当前弹窗中查找指定 ID 的元素。

标签式工作区(navtab)

B-JUI 框架的整个工作区部分就是一个 `navtab` 组件，本组件位于主页面的"#bjui-container"容器内，固定的 html 结构如下：

```

<div id="bjui-navtab" class="tabsPage">
    <div class="tabsPageHeader">
        <div class="tabsPageHeaderContent">
            <ul class="navtab-tab nav nav-tabs">
                <li data-tabid="main" class="main active"><a href="javascript:;"><span>我的主页
</span></a></li>
            </ul>
        </div>
        <div class="tabsLeft"><i class="fa fa-angle-double-left"></i></div>
        <div class="tabsRight"><i class="fa fa-angle-double-right"></i></div>
        <div class="tabsMore"><i class="fa fa-angle-double-down"></i></div>
    </div>
    <ul class="tabsMoreList">
        <li><a href="javascript:;">我的主页</a></li>
    </ul>
    <div class="navtab-panel tabsPageContent layoutBox">
        各页片内容区域
    </div>
</div>

```

用法(如何创建一个 `navtab`):

DOM 点击触发：

`打开`

```

navtab</a>
<button type="button" class="btn-green" data-toggle="navtab" data-id="mynavtab"
data-url="mytab.html" data-title="我的业务页面">打开 navtab</button>
jQuery:
$(selector).navtab({id:'标签的 id', url:'载入页片的 url', title:'标签的标题'})

```

取得当前 navtab 的内容容器: \$.CurrentNavtab

参数：

名称 类型 默认值 描述 id string navtab 标签的 ID，如果指定重复，将覆盖现有的 ID 相同标签。
title string New tab 标签打开后显示的名称。 url string undefined 请求数据的 url。 type string GET
Http 请求方式，可选 ‘GET/POST’。 data object {} 请求 url 时，需要发送的 data 数据。

方法：

这样调用 navtab 的方法：

```
$(selector).navtab('方法名', 参数 1, ..., 参数 n)
```

如要切换到某个标签：

```
$(selector).navtab('switchTab', 标签 ID)
```

方括号的参数，表示该参数可选。

可用的方法：

switchTab(tabid): 切换到某个标签。

refresh(tabid): 刷新某个标签。

reload(options): 重新载入某个标签，options 同 navtab 默认参数，如果未指定 ID，则默认重载入当前标签。

closeTab(tabid): 关闭某个标签。

closeCurrentTab([tabid]): 关闭当前标签。

closeAllTab(): 关闭所有标签。

事件：

这样监听 navtab 的事件：

```

$(document).on('bjui.beforeLoadNavtab', function(e) {
    var $navtab = $(e.target)
    // do something...
})

```

名称 描述 bjui.beforeLoadNavtab 载入标签内容前的事件 bjui.beforeCloseNavtab 关闭标签前的事件

弹出窗口(dialog)

弹出窗口分为普通弹出窗口和模态弹出窗口，普通弹出窗口可通过 taskBar 组件进行最小化等操作。弹出窗口的 DOM 结构会放入 Body 中，结构如下：

```

<div class="bjui-dialog bjui-dialog-container">
    <div class="dialogHeader">

```

```

        <!-- 最大化、最小化、关闭等按钮区 -->
        <h1><!-- 标题 --></h1>
    </div>
    <div class="dialogContent layoutBox unitBox">
        <!-- 页片内容区 -->
    </div>
    <!-- 用于调整大小的 div 片断 -->
</div>

```

用法(如何创建一个 dialog):

DOM 点击触发:

```

<a href="mydialog.html" data-toggle="dialog" data-id="mydialog" data-title="我的弹出窗口">打开弹出窗口</a>

```

```

<button type="button" class="btn-green" data-toggle="dialog" data-id="mynavtab" data-url="mytab.html" data-title="我的弹出窗口">打开弹出窗口</button>

```

jQuery:

```

$(selector).dialog({id:'弹窗的 id', url:'载入页片的 url', title:'弹窗的标题'})

```

取得当前 dialog: \$.CurrentDialog

参数:

名称 类型 默认值 描述 id string navtab 弹窗的 ID, 如果指定重复, 将覆盖现有的 ID 相同弹窗。 title string New tab 弹窗打开后显示的名称。 url string undefined 请求数据的 url。 type string GET Http 请求方式, 可选 'GET/POST'。 data object {} 请求 url 时, 需要发送的 data 数据。 width int 500 弹窗的宽度。 height int 300 弹窗的高度。 max boolean false 打开弹窗时直接最大化。 mask boolean false 是否模态窗口。 resizable boolean true 可以调整弹窗的大小。 drawable boolean true 可以拖动弹窗。 maxable boolean true 是否显示最大化按钮。 minable boolean true 是否显示最小化按钮 (模态弹窗无效)。

方法:

调用方式同 navtab: 如要关闭某个弹窗:

```

$(selector).dialog('close', 弹窗 ID)

```

方括号的参数, 表示该参数可选。

可用的方法:

switchDialog(id): 切换到某个弹窗(模态弹窗无效)。

refresh(id): 刷新某个弹窗。

reload(options): 重新载入某个弹窗, options 同 dialog 默认参数, 如果未指定 ID, 则默认重载入当前弹窗。

close(id): 关闭某个弹窗。

closeCurrent(): 关闭当前弹窗。

事件:

这样监听 dialog 的事件:

```

$(document).on('bjui.beforeLoadDialog', function(e) {

```

```

    var $dialog = $(e.target)
    // do something...
  })

```

名称 描述 `bjui.beforeLoadDialog` 载入弹窗内容前的事件 `bjui.beforeCloseDialog` 关闭弹窗前的事件

Ajax

本节主要介绍 B-JUI 框架中的 Ajax 操作，回调函数等。

Ajax 操作：

ajax 搜索 - 主要用于搜索表单，分页表单，例：B-JUI 源码"table.html"中的#pagerForm:

```
<form id="pagerForm" data-toggle="ajaxsearch" action="table.html" method="post">
```

ajax 重置搜索 - 主要用于重置搜索表单，`data-clear-query` 属性告诉 form 是否清空查询参数，仅保留分页及字段排序信息，例：B-JUI 源码"table.html"中的清空查询按钮：

```
<a class="btn btn-orange" href="javascript:;" data-toggle="reloadsearch" data-clear-query="true" data-icon="undo">清空查询</a>
```

ajax 载入 - 主要用于为指定容器载入 url 的内容，`data-target` 属性值为选择器表达式，告诉 ajax 载入的内容放到该容器，适合用来做局部刷新：

```
<a href="table-edit.html" data-toggle="ajaxload" data-target="#D11">加载内容</a>
```

ajax 动作 - 主要用于执行 ajax 命令，如“删除”，`data-confirm-msg` 属性用于在操作前进行确认提示：

```
<a href="del.html" class="btn btn-red" data-toggle="doajax" data-confirm-msg="确定要删除该行信息吗？">删除数据</a>
```

ajax 导出 - 主要用于导出信息(下载文件)，`data-confirm-msg` 属性用于在操作前进行确认提示：

```
<a href="book1.xlsx" data-toggle="doexport" data-confirm-msg="确定要导出信息吗？">导出全部</a>
```

ajax 导出选中项 - 主要用于导出选中项信息(下载文件)，`data-confirm-msg` 属性用于在操作前进行确认提示，`data-idname` 属性指定发送到后台的字段名称，默认“ids”，`data-group` 属性用于指定复选框的 name：

```
<a href="book1.xlsx" data-toggle="doexportchecked" data-confirm-msg="确定要导出选中项吗？" data-idname="expids" data-group="ids">导出选中项</a>
```

ajax 删除选中项 - 主要用于导出选中项信息(下载文件)，`data-confirm-msg` 属性用于在操作前进行确认提示，`data-idname` 属性指定发送到后台的字段名称，默认“ids”，`data-group` 属性用于指定复选框的 name：

```
<a href="del.html" data-toggle="dodelchecked" data-confirm-msg="确定要删除选中项吗？" data-idname="delids" data-group="ids">导出选中项</a>
```

Ajax 回调：

ajaxsearch、doajax、dodelchecked 三种 ajax 操作支持自定义 ajax 回调函数，自定义的回调函数放到 `data-callback` 属性，写法如下：。

`删除数据`

```
function mycallback(json) {  
    //do something...  
}
```

服务端返回的 JSON 参数如下：

名称 类型 描述 statusCode int 必选。状态码(ok = 200, error = 300, timeout = 301), 可以在 BJUI.init 时配置三个参数的默认值。 message string 可选。信息内容。 tabid string 可选。待刷新 navtab id, 多个 id 以英文逗号分隔开, 当前的 navtab id 不需要填写, 填写后可能会导致当前 navtab 重复刷新。 closeCurrent boolean 可选。是否关闭当前窗口(navtab 或 dialog)。 forward string 可选。跳转到某个 url。 forwardConfirm string 可选。跳转 url 前的确认提示信息。

默认的回调函数会根据当前触发元素的位置确定是刷新 navtab 还是 dialog, 或是局部 div 容器。

可调用的返回信息提示：

```
$(selector).bjuiajax('ajaxDone', json)
```

表格

表格的 Class 同 Bootstrap, 仅调整了 padding 值, 使表格显示得更紧凑。

.table 少量的 padding 和水平分隔线。

.table-striped 条纹状表格 [IE8 不支持]。

.table-bordered 带边框表格。

.table-hover 附加鼠标悬停效果。

表格表头字段排序按钮实现：

```
<th data-order-field="sex">姓名</th>
```

```
<th data-order-direction="asc" data-order-field="createtime">创建时间</th>
```

data-order-field="" 属性令本字段可排序, 属性值为与后台交互的字段名称。

data-order-direction 属性表示本字段的当前排序状态, 可选值(asc/desc)

普通表格

普通的表格, 可显示列表状条目, 或布局表单, 表头字段可附加排序按钮。

完整的实例详见 B-JUI 源代码的 table.html。

固定表头表格

表格的头可固定住, 可调整各列的宽度, 表头字段可附加排序按钮。

```
<table data-toggle="tablefixed" data-width="100%" data-layout-h="0">
```

表格添加属性 data-toggle="tablefixed" 后就会固定表头, B-JUI 默认为固定的表头的表格添加 Class: table table-striped table-bordered table-hover, 如果该 table 设置有 Class, 则以设置的为准 data-width 属性可定义固定表格的宽度, 默认为"100%", 可设置大于 100%或固定值, 如: 150% 和 1200 都是合法的, 超过 navtab 工作区的宽度时会出现横向滚动条。

完整的实例详见 B-JUI 源代码的 table-fixed.html。

可编辑表格

用于需要动态添加简单行内容的地方。

```
<table class="table table-bordered table-hover table-striped" data-toggle="tableedit" data-initnum="0"  
data-layout-h="0">
```

```

<thead>
  <tr>
    <th title="No."><input type="text" name="edit[#index#].id" class="no" data-rule="required"
value="1" size="2"></th>
    <th title="姓名"><input type="text" name="edit[#index#].name" data-rule="required" value=""
size="5"></th>
    <th title="" data-addtool="true" width="100">
      <a href="javascript:;" class="btn btn-red row-del" data-confirm-msg="确定要删除该行信息
吗？">删除</a>
    </th>
  </tr>
</thead>
</table>

```

实现步骤:

表格添加属性 data-toggle="tablefixed";

thead 中加上对应的表单信息，字段名放到 title 属性上；

若想在某列上显示添加按钮，需在对应列上添加属性 data-addtool="true";

若每行显示删除按钮，需为删除按钮添加 Classrow-del，如果要确认才能删除，需为删除按钮添加属性 data-confirm-msg="删除提示信息"。

表格的 data-initnum 属性，表示载入本页片时，初始化的添加行数，值为 0 时不需写。

外部触发添加行事件：

按钮式：

```

<button type="button" class="btn-green" data-toggle="tableeditadd" data-target="#tableedit1"
data-num="1">添加编辑行</button>

```

data-target 属性设置编辑表格的 jQuery 选择器，data-num 属性设置点击时添加的行数。

jQuery: \$(selector).tableedit('add', 编辑表格的 jQuery 对象, 待添加的行数)

\$(selector).tableedit('add', \$('#tableedit1'), 2)

完整的实例详见 B-JUI 源代码的 table-edit.html。