Atitit bootstrapTable的使用

目录

[1. 定义表格 1](#_Toc31702)

[1.1. 定义表头html 數據屬性模式和js模式 1](#_Toc18968)

[1.2. 通過JavaScript 2](#_Toc15185)

[1.3. 推送加载数据 4](#_Toc28761)

[2. 表格组件神器：bootstrap table详细使用指南 4](#_Toc8476)

[2.1. 5.1、当请求返回的数据结果不是固定的“total”,"rows"的格式时，如何渲染表格数据？ 5](#_Toc8561)

[2.2. 字段格式化时间 7](#_Toc8707)

[2.3. 增加了几个操作列 也是通过格式化函数 8](#_Toc12105)

[2.4. 表头对不齐 9](#_Toc24434)

[2.5. 表格删除 一行后对刷新操作 9](#_Toc30775)

[3. Other 10](#_Toc24216)

[3.1. 表头杂乱，使用二级表头，h5合并模式 10](#_Toc21528)

[3.2. 行操作传递对象，使用json序列化+特殊符号编码法 10](#_Toc2462)

[3.3. 字段过长，可以俩个字段内容合并，换行显示 10](#_Toc14123)

# 定义表格

<table class="table table-striped" id="dataTable">

</table>

当然Bootstrap table还提供了另外两种简单的用法，直接通过$table.bootstrapTable({data: data})，具体参考**form-data.html;**  
或者直接在table标签里面定义 <table data-toggle="table" data-url="../json/data1.json">,类似“data-...”等相关属性，具体参考**table-style.html;这两种方法都**不用在js里面注册就可以实现数据的加载，

但博主觉得这种用法虽然简单，但不太灵活，所以咱们还是统一使用在js里面初始化的方式来使用table组件。

## 定义表头html 數據屬性模式和js模式

Bootstrap Table插件通過數據屬性或JavaScript以表格形式顯示數據。

我們還可以通過data-url="data1.json"在普通表上進行設置來使用遠程URL數據。

複製

<table

data-toggle="table"

data-url="data1.json">

<thead>

<tr>

<th data-field="id">Item ID</th>

<th data-field="name">Item Name</th>

<th data-field="price">Item Price</th>

</tr>

</thead></table>

您還可以將pagination，search和添加sorting到下表中的表格。

複製

<table

data-toggle="table"

data-url="data1.json"

data-pagination="true"

data-search="true">

<thead>

<tr>

<th data-sortable="true" data-field="id">Item ID</th>

<th data-field="name">Item Name</th>

<th data-field="price">Item Price</th>

</tr>

</thead></table>

## 通過JavaScript

通過JavaScript調用帶有id表的引導表。

複製

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

複製

$('#table').bootstrapTable({

columns: [{

<table class="table table-striped" id="dataTable">

<thead>

<tr>

<th colspan="2" data-valign="middle" data-halign="center">用户基本信息</th>

<th colspan="4" data-valign="middle" data-halign="center">补单信息</th>

<th colspan="4" data-valign="middle" data-halign="center" data00-formatter00="operateFormatter">其他信息</th>

</tr>

<tr>

<th data-field="userId" data-align="center" data-formatter="setCode">用户编号</th>

<th data-field="account" data-align="center">姓名</th>

<th data-field="operationTime" data-align="center">时间</th>

<th data-field="changeMoney" data-align="center" data-sortable="true">金额</th>

<th data-field="operationType" data-align="center">上分下分</th>

<th data-field="gameType" data-align="center">gameType</th>

<th data-field="pk" data-align="center">pk</th>

<th data-field="orderNumber" data-align="center">orderNumber</th>

<th data-field="errorCount" data-align="center">自动补单次数</th>

<th data-field="operationType" data-align="center">operationType</th>

</tr>

</thead>

</table>

## 推送加载数据

  $('#' + tableName).empty();

   $('#' + tableName).bootstrapTable(refresh).bootstrapTable({

**5.3.2、使用ajax调用接口获取数据，再通过$('#tableName').bootstrapTable({ data: data})的方式，实现客户端排序；**

# [表格组件神器：bootstrap table详细使用指南](https://www.cnblogs.com/wdlhao/p/6694083.html)

2017-04-12 09:06  [流浪的诗人](https://www.cnblogs.com/wdlhao/)  阅读(64640)  评论(35)  [编辑](https://i.cnblogs.com/EditPosts.aspx?postid=6694083)  [收藏](https://www.cnblogs.com/wdlhao/p/javascript:void(0))

1、bootstrap-table简介

* **1.1、bootstrap table简介及特征：**

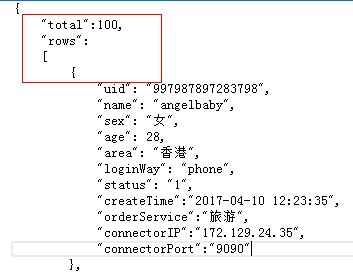
         Bootstrap table是国人开发的一款基于 Bootstrap 的 jQuery 表格插件，通过简单的设置，就可以拥有强大的单选、多选、排序、分页，以及编辑、导出、过滤（扩展）等等的功能。目前在github上已经有2600多个Star，可见其受欢迎程度。其官方网站地址 为：http://bootstrap-table.wenzhixin.net.cn/。里面可以下载使用所需的JS和CSS文件，以及参考文档和例子。

* 支持 Bootstrap 3 和 Bootstrap 2
* 自适应界面
* 固定表头
* 非常丰富的配置参数
* 直接通过标签使用
* 显示/隐藏列
* 显示/隐藏表头
* 通过 AJAX 获取 JSON 格式的数据
* 支持排序
* 格式化表格
* 支持单选或者多选
* 强大的分页功能
* 支持卡片视图
* 支持多语言
* 支持插件

5、遇到的问题及解决方法

## ****5.1、当请求返回的数据结果不是固定的“total”,"rows"的格式时，如何渲染表格数据？****

　　  我们运用$('#' + tableName).bootstrapTable({ url: '../data/login\_info.json'}),请求json数据，必须要求json数据格式为固定的key值，即必须按照“total”,"rows"的格式才能填充数据到表格，如图所示：



但是，当接口返回的数据格式，不是固定的“total”,"rows"时，我们需要用到一个转换函数responseHandler，将数据指引到“total”,"rows"上去。  
即用login\_info2.json和参数responseHandler实现数据填充。

$('#dataTable').bootstrapTable({

method: 'post',

url: tableurl,

responseHandler:responseHandlerRetDataFmt,

// trans ret data fmt

function responseHandlerRetDataFmt(res){

return {"rows":res,"total":999};

}

## 字段格式化时间

<th data-field="operationTime" data-align="center" data-formatter="dateFormat99">时间</th>

<script type="text/javascript">

function dateFormat99(value, row, index)

{

return changeDateFormat(value)

}

function operateFormatter(value, row, index)

{

if(value==0) return "》》上分";if(value==1) return "下分";

}

//转换日期格式(时间戳转换为datetime格式)

function changeDateFormat(cellval) {

var dateVal = cellval + "";

if (cellval != null) {

var date = new Date(parseInt(dateVal.replace("/Date(", "").replace(")/", ""), 10));

var month = date.getMonth() + 1 < 10 ? "0" + (date.getMonth() + 1) : date.getMonth() + 1;

var currentDate = date.getDate() < 10 ? "0" + date.getDate() : date.getDate();

var hours = date.getHours() < 10 ? "0" + date.getHours() : date.getHours();

var minutes = date.getMinutes() < 10 ? "0" + date.getMinutes() : date.getMinutes();

var seconds = date.getSeconds() < 10 ? "0" + date.getSeconds() : date.getSeconds();

return date.getFullYear() + "-" + month + "-" + currentDate + " " + hours + ":" + minutes + ":" + seconds;

}

}

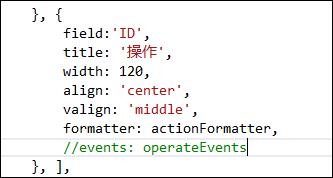
</script>

## 增加了几个操作列 也是通过格式化函数

另外，我们看到行记录的最后增加了几个操作按钮 也是通过格式化函数，方便对当前记录的查看、编辑和删除操作，如下效果图所示。



这部分我们也是通过格式化函数进行处理的



IMG_258

1 //操作栏的格式化

2 function actionFormatter(value, row, index) {

3 var id = value;

4 var result = "";

5 result += "";

6 result += "";

7 result += "";

8

9 return result;

10 }

## 表头对不齐

不要设置height即可。。

## 表格删除 一行后对刷新操作

$("#dataTable").bootstrapTable('refresh', {});

function dltRec(pk){

if(confirm("确定删除此单：pk:"+pk))

{

//alert( window.location.host) in file mode test ,is empty

delurl=basepathx+"budan/ajaxBudanDlt.json?pk="+pk;

$.get(delurl, function(result){

alert(result);

// $('#dataTable').empty();

$("#dataTable").bootstrapTable('refresh', {});

});

}

// alert(pk)

}

# Other

## 表头杂乱，使用二级表头，h5合并模式

## 行操作传递对象，使用json序列化+特殊符号编码法 base64 urlencode

base64编码 urlencode等

## 字段过长，可以俩个字段内容合并，换行显示

合并内容使用自定义格式化模式