[PHP]\_TP5.0\_FastAdmin\_组件；

--王尔贝(我是害虫)

(整理 <https://doc.fastadmin.net/docs/component.html>)

--

FastAdmin默认集成了多个第三方组合，如表单验证、文件上传、下拉列表、时间选择、城市选择、Selectpage，所有的组件都必须使用Form.api.bindevent("form[role=form]")来进行初始化，如果不进行初始化是无法对相应组件进行渲染和事件绑定。

//

// 如果我们想单独为元素绑定事件和渲染，我们可以使用如下的代码

组件 代码 介绍

文件上传 Form.events.plupload($("form")); //渲染并绑定form中的上传组件;

动态下拉列表 Form.events.selectpage($("form")); //渲染并绑定form中的Selectpage组件;

表单验证 Form.events.validator($("form")); //渲染并绑定form中的表单验证;

城市选择 Form.events.citypicker($("form")); //渲染并绑定form中的城市选择组件;

日期时间 Form.events.datetimepicker($("form")); //渲染并绑定form中的日期时间组件;

下拉列表 Form.events.selectpicker($("form")); //渲染并绑定form中的Selectpicker组件;

附件选择 Form.events.faselect($("form")); //渲染并绑定form中的选择附件组件;

键值组件 Form.events.fieldlist($("form")); //渲染并绑定form中的键值组件;

开关组件 Form.events.switcher($("form")); //渲染并绑定form中的开关组件;

滑块组件 Form.events.slider($("form")); //渲染并绑定form中的滑块组件;

--

**一、文件上传**

FastAdmin支持将文件或图片直传到第三方云存储服务器而不需要通过服务器进行中转，你可以直接在后台插件管理安装第三方云存储的插件后使用，目前支持以下云储存平台。

平台 特点 插件下载；

又拍云 申请加入联盟可享每月免费15G流量、图片处理 下载(<https://www.fastadmin.net/store/upyun.html>)；

七牛云 实名认证后免费10G流量、稳定、图片处理 下载(<https://www.fastadmin.net/store/qiniu.html>)；

阿里OSS 阿里系、稳定、图片处理、支持挂载为分区 下载(<https://www.fastadmin.net/store/alioss.html>)；

Ucloud 每月20G免费流量、图片处理 下载(<https://www.fastadmin.net/store/ucloud.html>)；

腾讯COS 每月50G免费流量、图片处理 下载(<https://www.fastadmin.net/store/cos.html>)；

//

在使用第三方云存储功能之前请先注册一个账号并新增一个云储存服务,你可以通过FastAdmin的邀请链接注册，FastAdmin会获得相应平台的CDN流量。

又拍云：https://www.fastadmin.net/go/upyun

七牛云：https://www.fastadmin.net/go/qiniu

阿里云：https://www.fastadmin.net/go/aliyun

腾讯云：https://www.fastadmin.net/go/tencent

//

FastAdmin的上传功能非常强大，我们只需要简单的配置即可支持单图或多图上传。

<div class="form-group">

<label for="c-avatar" class="control-label col-xs-12 col-sm-2">头像:</label>

<div class="col-xs-12 col-sm-8">

<!--文件上传-->

<div class="input-group">

<input id="c-avatar" data-rule="" class="form-control" size="50" name="row[avatar]" type="text" value="{$row.avatar}">

<div class="input-group-addon no-border no-padding">

<span><button type="button" id="plupload-avatar" class="btn btn-danger plupload" data-input-id="c-avatar" data-mimetype="image/gif,image/jpeg,image/png,image/jpg,image/bmp" data-multiple="false" data-preview-id="p-avatar"><i class="fa fa-upload"></i> 上传</button></span>

<span><button type="button" id="fachoose-avatar" class="btn btn-primary fachoose" data-input-id="c-avatar" data-mimetype="image/\*" data-multiple="false"><i class="fa fa-list"></i> 选择</button></span>

</div>

<span class="msg-box n-right" for="c-avatar"></span>

</div>

<ul class="row list-inline plupload-preview" id="p-avatar"></ul>

</div>

</div>

//

//// 组件元素：配置一个文本框、一个上传按钮、一个选择按钮、一个预览的DIV。

文本框 主要用于接收上传后返回的图片链接，文本框id属性是必选的，这里的id是c-avatar。

上传按钮 主要用于点击后上传文件，有几个属性非常重要，请参考下方的上传按钮属性介绍。

选择按钮 主要用于点击后选择已经上传的文件，有几个属性非常重要，请参考下方的选择按钮属性介绍。

预览区域 主要用于预览上传或选择文件后的预览，id属性是改造的，这里的id是p-avatar。

//

//// 上传按钮支持属性

属性 示例值 说明。

data-url ajax/upload 用于配置上传文件接收的地址。

data-multipart {"key1":"value1"} 用于上传时附加额外的参数信息。

data-input-id c-avatar 用于填充返回URL地址的设文本框。

data-mimetype image/gif,image/jpeg,image/png,image/jpg,image/bmp 用于过滤允许上传的文件类型，支持mimetype或文件后缀名。

data-multiple false 是否支持多图或多文件模式。

data-preview-id p-avatar 用于预览返回URL地址的DIV。

data-maxsize 10M 用于限制最大可上传的文件大小。

//

//// 选择按钮支持属性(属性 示例值 说明)

data-input-id c-avatar 用于填充返回URL地址的设文本框。

data-mimetype image/gif,image/jpeg,image/png,image/jpg,image/bmp 用于过滤允许上传的文件类型，支持mimetype或文件后缀名。

data-multiple false 是否支持多图或多文件模式。

data-preview-id p-avatar 用于预览返回URL地址的DIV。

//

如果我们想直接通过JS上传一个文件到我们的服务器，我们可以使用Upload.api.send(file, success, failure, complete)来上传文件。

--

**二、动态下拉列表(SelectPage)**

FastAdmin中的动态下拉列表使用的是优秀强大的Selectpage插件来支持，FastAdmin对其进行了二次开发。下面介绍一个最基础的动态下拉列表示例，如下所述。

<input id="c-name" data-rule="required" data-source="category/selectpage" class="form-control selectpage" name="row[name]" type="text" value="">

其中需要给元素class添加一个selectpage，其次需要增加一个data-source="category/selectpage"这个属性，category/selectpage为我们控制器提交列表的方法。FastAdmin的Selectpage列表中显示字段默认读取的是name字段，如果我们返回的列表中不包含name字段，将无法展现下拉列表数据。此时我们需要添加使用data-field="你要显示的字段"即可。FastAdmin的Selectpage列表中主键字段默认读取的是id字段，如果我们的主键不是id字段，则我们可以添加并使用data-primary-key="你的主键ID字段"来修改。

//

//// Selectpage扩展属性(属性 功能 示例)

data-source 提供数据源的URL地址 data-source="category/index"。

data-field 列表显示读取的字段 data-field="username"。

data-primary-key 列表选中后渲染的字段 data-primary-key="uid"。

data-pagination 是否开启分页 data-pagination="true"。

data-page-size 分页大小 data-page-size="10"。

data-multiple 是否支持多选 data-multiple="true"。

data-max-select-limit 最多可选择数量 data-max-select-limit="3"。

data-order-by 排序字段 data-order-by="id"。

data-params 自定义扩展参数 data-params='{"custom[type]":"test"}'。

data-select-only 是否为只读模式 data-select-only="true"。

//

Selectpage的data-params支持function类型,如果需要动态传参(例如联动查询),则可以在JS中将data-params添加一个function处理即可。FastAdmin在生成CRUD时会对包含下划线的字段默认生成动态下拉列表，比如user\_id将自动生成data-source="user/index"，默认读取的是id和name字段，如果需要修改，请参考上方的参数修改方法。

//

data-params自定义扩展参数支持使用function动态返回数据，请在表单初始化之前使用，例如传递动态选择的类型。

$("#c-name").data("params", function (obj) {

return {custom: {type: $("#c-type").val()}};

});

更多的使用方法请参考Selectpage官方教程(<https://terryz.github.io/>)。

--

**三、富文本编辑器**

FastAdmin的富文本编辑器只需要给对应的textarea增加一个class为editor即可，FastAdmin在渲染时即会将textarea渲染为富文本编辑器，目前支持summernote、nkeditor和ueditor等富文本编辑器，需安装对应的插件即可正常使用。

//

//// 插件 特点 插件下载

Summernote 操作简单、易用、图片直传第三方存储 下载(<https://www.fastadmin.net/store/summernote.html>)。

Nkeditor 基于Kindeditor、多主题、简洁强大、图片直传第三方存储 下载(<https://www.fastadmin.net/store/nkeditor.html>)。

Wangeditor 轻量、简洁、易用、图片直传第三方存储 下载(<https://www.fastadmin.net/store/wangeditor.html>)。

Simeditor 漂亮、简单、功能简单、图片直传第三方存储 下载(<https://www.fastadmin.net/store/simeditor.html>)。

Umeditor 百度出品、简单、易用、支持公式、图片直传第三方存储 下载(<https://www.fastadmin.net/store/umeditor.html>)。

Ueditor 百度出品、功能全、图片直传第三方存储 下载(<https://www.fastadmin.net/store/ueditor.html>)。

//

如果是自定义表单,例如在index页面,切记使用Form.api.bindevent绑定事件才会生效,否则单独添加个editor是不会生效的。在安装完对应的富文本插件后切记启用、刷新插件缓存并清除浏览器缓存后才生效。请务必同时只启用一个富文本编辑器，如果需要不同的页面展示不同的编辑器，需要修改编辑器插件代码，修改对应的选择器，将editor改成自定义的class。

--

**四、表单验证**

FastAdmin的表单验证采用的是Nice-validator验证插件，Nice-validator是一款非常强大的表单验证插件，通过简单在元素上配置规则，即可达到验证的效果。在FastAdmin当中我们只需要给元素添加data-rule="规则"即可开启Nice-validator的验证，常用的规则如下所述。

//

//// 规则 描述 示例

required 字段必填 required。

checked 必选，只适用于checkbox和radio checked。

match(name) 当前字段值必须和 name 字段的值匹配 match('row[username]')。

remote(URL) 请求服务端验证 remote('validate/check\_username\_unique')。

integer 整数 integer。

range(n~) 数值范围, 请填写不小于 n 的数 range(3~)。

length(n) 请填写 n 个字符 length(3)。

filter 过滤 <>`"' 和字符实体编码的字符 filter。

digits 必须为数字 digits。

letters 必须为字母 letters。

date 必须为日期,yyyy-mm-dd格式 date。

time 必须为时间,hh:ii格式 time。

email 必须为email格式 email。

url 必须为URL链接 url。

qq 必须为QQ号 qq。

IDcard 必须为身份证号码 IDcard。

tel 必须为电话号码 tel。

mobile 必须为手机号码 mobile。

zipcode 必须为邮箱 zipcode。

chinese 必须为中文字符 chinese。

username 必须为3-12位数字、字母、下划线 username。

password 必须为6-16位字符，不能有空格 password。

//

如果你的表单不希望使用nice-validator组件进行验证，可以给form添加一个novalidate属性即可。更多的使用方法请参考Nice-validator官方教程(<https://validator.niceue.com/docs/>)。

--

**五、城市选择**

FastAdmin中集成了强大的city-picker城市选择插件，可以很方便的选择省份和城市。我们只需要简单的为input元素添加一个data-toggle="city-picker"属性即可自动渲染相应的城市选择组件

//

//// 城市选择属性(属性 描述 示例)

data-level 选择城市的级别，支持province/city/district，默认为district data-level="city"。

data-simple 使用简单的地址，比如使用内蒙古替代内蒙古自治区，默认为false data-simple="true"。

data-responsive 是否为自适应，默认为false data-responsive="true"。

placeholder 默认提示的文字 placeholder="请选择省/市"。

//

city-picker组件默认选择后渲染的是中文城市信息，我们可以通过在JS中监听city-picker更新后的事件来获取省份城市地区对应的code值。代码如下所述。

$("#city-picker").on("cp:updated", function() {

var citypicker = $(this).data("citypicker");

var code = citypicker.getCode("district") || citypicker.getCode("city") || citypicker.getCode("province");

$("#code").val(code);

});

如果我们数据库中存放的是地区的code值，在显示时我们则需要把对应的code通过读取数据库转换成我们的地区中文，然后再设定input的value值即可。更多的使用方法请参考city-picker官方教程(<https://github.com/tshi0912/city-picker>)。

--

**六、日期时间**

在FastAdmin中的日期时间组件采用的是Bootstrap-datetimepicker插件。我们在使用只可以为文本框添加一个class为datetimepicker的值即可自动添加日期时间选择框。同时我们还可以通过配置以下属性来自定义我们日期选择器的功能。

//

//// 属性 描述 示例

data-date-format 日期时间的格式，支持Moment.js的格式 data-date-format=" YYYY-MM-DD"。

data-date-min-date 最小可选择的日期 data-date-min-date="2011-10-01"。

data-date-max-date 最大可选择的日期 data-date-max-date="2046-10-01"。

data-date-use-current 使用当前的日期时间 data-date-use-current="true"。

data-date-default-date 默认日期 data-date-default-date="2011-10-01"。

data-date-disabled-dates 禁用的日期组 data-date-disabled-dates='["2011-10-02"]'。

data-date-enabled-dates 可用的日期组 data-date-enabled-dates='["2011-10-02"]'。

data-date-use-strict 使用严格的日期时间,错误日期将被忽略 data-date-use-strict="true"。

data-date-side-by-side 日期时间并排显示 data-date-side-by-side="true"。

//

更多的使用方法请参考Bootstrap-datetimepicker官方教程(<https://eonasdan.github.io/bootstrap-datetimepicker/>)。

--

**七、下拉列表**

在FastAdmin中集成了Bootstrap-select插件，可以对原有的select元素重新渲染，并增加相应的功能。我们可以直接给select元素添加一个class为selectpicker的值即可，FastAdmin在检测到以后会自动进行渲染，我们同时可以给select添加以下属性用于配置selectpicker。

//

//// 属性 介绍 添加位置 示例

data-live-search 是否启用动态搜索 select data-live-search="true"。

data-tokens 添加搜索的关键字 option data-tokens="keyword keyword2"。

data-max-options 最大可选择option的数量 select\optgroup data-max-options="2"。

title 自定义默认提示语 select title="请选择相应的分类"。

title 自定义选中以后显示的文字 option title="分类1"。

data-style 定义样式 select data-style="btn-primary"。

//

更多的使用方法请参考：Selectpicker官方教程(<https://silviomoreto.github.io/bootstrap-select/>)。

--

**八、键值组件**

键值组件是FastAdmin开发的一项简洁实用的基础组件，在FastAdmin中很多模块都有使用到该组件，例如常规管理->系统配置->字典配置均使用此组件开发，我们在插件管理配置中也经常可以看到键值组件的身影。

//

// 键值组件最常用使用方法

<dl class="fieldlist" data-name="row[configgroup]">

<dd>

<ins>键名</ins>

<ins>键值</ins>

</dd>

<dd>

<a href="javascript:;" class="btn btn-sm btn-success btn-append"><i class="fa fa-plus"></i> 追加</a>

</dd>

<textarea name="row[configgroup]" class="form-control hide" cols="30" rows="5">{"basic":"基础配置","email":"邮件配置","dictionary":"字典配置","user":"会员配置","example":"示例分组"}</textarea>

</dl>

通过将以上代码放置在我们的表单中，然后使用Form.api.bindevent("form")或Form.events.fieldlist("form")进行初始化即可。

//

以上是最简洁的使用方法，fieldlist还有更强大的自定义功能，如下所述。

<dl class="fieldlist" data-name="row[test]" data-template="testtpl">

<dd>

<ins>姓名</ins>

<ins>性别</ins>

<ins>年龄</ins>

<ins>成绩</ins>

</dd>

<dd>

<a href="javascript:;" class="btn btn-sm btn-success btn-append"><i class="fa fa-plus"></i> 追加</a>

</dd>

<textarea name="row[test]" class="form-control hide" cols="30" rows="5">[{"name":"张三"},{"gender":"男"},{"age":"23"},{"score":"80"}]</textarea>

</dl>

<!--定义模板-->

<script type="text/javascript" id="testtpl">

<dd class="form-inline">

<input type="text" name="row[<%=name%>][<%=index%>][name]" class="form-control" value="<%=row['name']%>" size="10">

<input type="text" name="row[<%=name%>][<%=index%>][gender]" class="form-control" value="<%=row['gender']%>" size="30">

<input type="text" name="row[<%=name%>][<%=index%>][age]" class="form-control" value="<%=row['age']%>" size="30">

<input type="text" name="row[<%=name%>][<%=index%>][score]" class="form-control" value="<%=row['score']%>" size="30">

<span class="btn btn-sm btn-danger btn-remove"><i class="fa fa-times"></i></span>

<span class="btn btn-sm btn-primary btn-dragsort"><i class="fa fa-arrows"></i></span>

</dd>

</script>

通过以上定义，可以任意自定义我们展示项的数据。

//

如果我们需要在点击追加按钮以后再对新增的展示项绑定事件，我们可以在JS中通过监听事件来给新增的元素绑定事件。

$(document).on("fa.event.appendfieldlist", 'data-name="row[test]"', function(){

Form.api.bindevent(this);

});

--

**九、开关组件**

开关组件常用于状态值的变更或只有两个值的切换。使用开关组件只需要给我们的操作按钮添加data-toggle="switcher"即可，如下所述。

<input id="c-switch" name="row[switch]" type="hidden" value="0">

<a href="javascript:;" data-toggle="switcher" class="btn-switcher" data-input-id="c-switch" data-yes="1" data-no="0" >

<i class="fa fa-toggle-on text-success {eq name="$row.switch" value="0"}fa-flip-horizontal text-gray{/eq} fa-2x"></i>

</a>

--