App.js的api v1.3

[1 外部函数api 1](#_Toc420609722)

[1.1 init() 1](#_Toc420609723)

[1.2 initAjax() 1](#_Toc420609724)

[1.3 initPlugins() 2](#_Toc420609725)

[1.4 fixContentHeight() 2](#_Toc420609726)

[1.5 addResponsiveHandler() 2](#_Toc420609727)

[1.6 scrollTo(el, offeset) 3](#_Toc420609728)

[1.7 scrollTop() 3](#_Toc420609729)

[1.8 blockUI(el, centerY) 4](#_Toc420609730)

[1.9 unblockUI(el) 4](#_Toc420609731)

[1.10 getActualVal(el) 4](#_Toc420609732)

[1.11 isIE8() 5](#_Toc420609733)

[1.12 isIE9() 5](#_Toc420609734)

[1.13 ajaxToPage(url) 6](#_Toc420609735)

[1.14 ajaxToModal(url) 6](#_Toc420609736)

[1.15 ajaxSubmit(url) 7](#_Toc420609737)

[1.16 ajaxSubmitModal(url) 7](#_Toc420609738)

[1.17 alert(options) 8](#_Toc420609739)

[1.18 alertSuccess(message, selector, isFocus) 9](#_Toc420609740)

[1.19 alertDanger(message, selector, isFocus) 9](#_Toc420609741)

[1.20 toastSuccess(message, callback) 10](#_Toc420609742)

[1.21 toastError(message, callback) 10](#_Toc420609743)

[1.22 post(url, data, callback) 11](#_Toc420609744)

[1.23 postModal(url, data, callback) 11](#_Toc420609745)

[2 class选择器监听事件 12](#_Toc420609746)

[2.1 侧边栏跳转监听 12](#_Toc420609747)

[2.2 内页（page-content-body）中跳转监听 12](#_Toc420609748)

[2.3 tree-table插件初始化 12](#_Toc420609749)

[2.4 date-picker插件初始化 12](#_Toc420609750)

[2.5 ueditor插件初始化 13](#_Toc420609751)

# 1 外部函数api

## 1.1 init()

初始化系统布局、组件及一些基本事件，在index初始化页面时调用一次即可。一般情况不会调用。

**返回：**

无

## 1.2 initAjax()

在Ajax请求页面后初始化页面内的一些组件及其事件，在Ajax请求后的callback中调用。一般情况不会调用。

**返回：**

无

## 1.3 initPlugins()

在Ajax请求页面后初始化页面内引入的的第三方插件，在页面require请求后的callback中调用。

**返回：**

无

## 1.4 fixContentHeight()

在Ajax请求页面后调整侧边栏和内容相应的高度，在Ajax请求后的callback中调用。一般情况不会调用。

**返回：**

无

## 1.5 addResponsiveHandler()

添加页面响应式变化后的回调事件，一般情况不会调用。

**返回：**

无

## 1.6 scrollTo(el, offeset)

使页面滚动到el元素的位置，offeset是el的纵向相对位移。

**返回：**

无

## 1.7 scrollTop()

使页面滚动到顶端。

**返回：**

无

## 1.8 blockUI(el, centerY)

给el元素添加等待遮罩层，centerY为布尔值，表示是否在纵向居中，一般填写false。一般在发送ajax请求或处理复杂操作时调用。

**返回：**

无

## 1.9 unblockUI(el)

给el元素移除等待遮罩层，调用blockUI后记得相应调用。

**返回：**

无

## 1.10 getActualVal(el)

获得el元素中真实的值（解决placeholder属性在IE8,IE9中不被支持的问题）。

**返回：**

el元素中的字符串值

## 1.11 isIE8()

判断是否IE8。

**返回：**

布尔值

## 1.12 isIE9()

判断是否IE9。

**返回：**

布尔值

## 1.13 ajaxToPage(url)

在内页（page-content-body）中使用ajax请求url返回的页面，用于发送ajax请求成功后进行页面跳转，请求失败则返回错误信息显示的操作。

**返回：**

无

## 1.14 ajaxToModal(url)

在模态框（modal）中使用ajax请求url返回的页面，用于发送ajax请求成功后进行modal页面的刷新，请求失败则返回错误信息显示的操作。

**返回：**

无

## 1.15 ajaxSubmit(url)

在内页（page-content-body）中使用ajax提交表单，提交表单请求成功后进行页面跳转，请求失败则返回错误信息显示。在标签中配置表单验证会自动进行验证。

**返回：**

无

## 1.16 ajaxSubmitModal(url)

在模态框（modal）中使用ajax提交表单，提交表单请求成功后进行modal页面的刷新，请求失败则返回错误信息显示。在标签中配置表单验证会自动进行验证。

**返回：**

无

## 1.17 alert(options)

弹出提醒框：可选配置参数options：

{

container: "", // 父容器(默认在.page-content容器的头部)

place: "prepend", // 在头部或者尾部弹出，取值范围：append，prepend（在指定了container时生效）

type: 'success',  // 提醒类型（取值范围：success，info，warning，danger）

message: "",  // 提醒内容

close: true, // 可否手动关闭

reset: true, // 提醒前是否清楚上一个提醒

focus: true, // 焦点是否转移到提醒处

closeInSeconds: 30, // 是否设定自动关闭（指定为0则不自动关闭）

icon: "" // 在提醒内容前设置图标（如成功check，警告warning）

}

**返回：**

无

## 1.18 alertSuccess(message, selector, isFocus)

默认提供弹出成功的提醒框，message为必填提醒信息，selector为提醒显示的容器选择器，isFocus配置焦点是否转移到提醒处。省略selector则默认显示在页面顶端，isFocus默认为false；此方法已被toastSuccess替代。

**返回：**

无

## 1.19 alertDanger(message, selector, isFocus)

默认提供弹出失败的提醒框，message为必填提醒信息，selector为提醒显示的容器选择器，isFocus配置焦点是否转移到提醒处。省略selector则默认显示在页面顶端，isFocus默认为false；此方法已被toastError替代。

**返回：**

无

## 1.20 toastSuccess(message, callback)

默认提供在窗口头部弹出成功的提醒浮动框，2秒消失，message为必填提醒信息，callback为点击浮动框后触发的回调函数。

**返回：**

无

## 1.21 toastError(message, callback)

默认提供在窗口头部弹出失败的提醒浮动框，4秒消失，message为必填提醒信息，callback为点击浮动框后触发的回调函数。

**返回：**

无

## 1.22 post(url, data, callback)

在内页（page-content-body）中使用post方式异步请求，data为json格式的参数，callback是成功回调函数，为可选参数，省略则默认执行页面跳转。

**返回：**

无

## 1.23 postModal(url, data, callback)

在模态框（modal）中使用post方式异步请求，data为json格式的参数，callback是成功回调函数，为可选参数，省略则默认执行modal中的页面跳转。

**返回：**

无

# 2 class选择器监听事件

## 2.1 侧边栏跳转监听

侧边栏a标签"ajaxify"的class，请求的url写在url属性中。

## 2.2 内页（page-content-body）中跳转监听

支持所有标签"ajaxify"的class，请求的url写在url属性中。

## 2.3 tree-table插件初始化

table标签加上"tree-table"的class即可初始化为树状表格。

## 2.4 date-picker插件初始化

input标签加上"date-picker"的class即可初始化为日期选择控件。

## 2.5 ueditor插件初始化

textarea标签加上"ueditor"的class即可初始化为富文本控件。