前端模板使用说明

# 公共函数

## WidgetCalls 类说明

WidgetCalls是一个函数登记调用机制，起到分离函数调用和函数定义的目的。

WidgetCalls作用在DOM节点上，以DOM节点和在此节点命名空间内的功能识别字为单位登记和调用函数，每个识别字可以登记多个函数，执行时会按照登记顺序逐一执行登记的函数，第一个返回的结果成为下一个函数的参数，最后一个函数的返回结果成为回调函数（如果回调函数不为空）的参数。

属性

方法

execute执行定义的函数（集）

参数：

id：DOM节点id

action：动作识别号

data：传递个函数的参数数据

callback：回调函数

add增加函数，为id和action增加一个函数，相同的id呵action可以增加多个函数

参数：

id：DOM节点id

action：动作识别号

func：函数

addDefault增加函数，为id和action增加一个缺省函数函数，当没有为此节点和识别号增加函数是调用此函数

参数：

id：DOM节点id

action：动作识别号

func：函数

createNew（）创建WidgetCalls实例

例子:

var mc = createNew()；

func=function(obj){

var id=obj.id;

var action=obj.action;

var data = obj.data;

console.log(id,action,data); /\* div1 abc data1 \*/

};

mc.add(‘div1’,’abc’,func);

mc.execute(‘div1’,’abc’,’data1’,null); /\* callback函数为空 \*/

# tmplRender

tmplRender模板翻译，将数据和模板翻译成html代码

格式 tmplRender(ctmplName,data)

参数：

ctmplName：模板名称，必须已经存放在/ctmpl中的前端模板，名字中不包含前端模板名称的后缀”.ctmpl”

data：{}格式的数据

例子：

var menudata={

“id”:”div1”,

“submenu”:[

{

“label”:”菜单项1”,

“icon”:”icon-save”

},{

“label”:”菜单项2”,

“icon”:”icon-edit”,

“submenu”:[

{

“label”:”菜单项3”

},{

“label”:”菜单项3”

}

]

}

]

};

var d = tmplRender(‘menu’,menudata);

# 前端模板

## accordion

数据格式要求：

accordion数据

{

“id”:”id1”,

“width”:”200px”,

“height”:”100%”,

“items”:[

{

“label”:”title1”,

“icon”:”icon-save”,

“url”:”/tmp1/test1.html”

},

{

“label”:”title2”,

“icon”:”icon-save”,

“url”:”/tmp1/test2.html”

},

{

“label”:”title3”,

“icon”:”icon-save”,

“url”:”/tmp1/test3.html”

},

}

Datagrid:

{

"id":"data1",

"url":"getData.url",

“icon”:”icon-save”,

"appendURL":"append.url",

"deleteURL":"delete.url",

"updateURL":"update.url",

"idField":"gg",

”dnd”:true,

"fields":[

{

"name":"gg",

"iotype":"text",

"label":"grage"

},

{

"name":"gg1",

"iotype":"date",

"label":"birthdate"

},

{

"name":"gg",

"iotype":"urlcodes",

"label":"grage"，

“textField”:”name”,

“valueField”:”code”

},

{

"name":"gggg",

"iotype":"codes",

"label":"grage"，

“data”:[

{

“value”:”1”,

“label”:”男“,

},

{

“value”:”0”,

“label”:”女“,

},

]

}

]

}

Iotype:支持以下类型

“date”

“datetime”,

“codes”,

“urlcodes”

“textbox”,

槽：

onRowContextMenu 手标右键按钮

onClickRow 点击记录

onClickCell 点击单元格

onSelect 记录已选择

onDblClickRow 记录双击

onDblClickCell 单元格双击

# toolbar

toolbar数据要求

{

“id”:”id1”,

“tools”:[

{

“name”:”new”,

“icon”:”icon-new”,

“text”:”新建“

},{

“name”:”edit”,

“icon”:”icon-edit”,

“text”:”编辑“

},{

“name”:”save”,

“icon”:”icon-save”,

“text”:”保存“

}

]

}

槽：

Tools中的每个name都是一个槽

## Layout

{

“id”:”toplayout”,

“width”:

}

要使用前端模板，需要在”header.tmpl”中包含”clientTemplate.tmpl”模板：{% include ”clientTemplate.tmpl” %}，和”clientTemplate.js”

前端模板必须放在后台网站的/ctmpl目录下面。

目前支持的前端模板有：

accordion,datagrid,datalist,echarts\_single,form,layout,menu,pie,tabs,toolbar,tree,treegrid

模板开发者

应用模板者

treegrid模板

提供以下数据便可以生成treegrid。

{

"\_\_ctmpl\_\_":"treegrid",

"data":{

"id":"testtg",

"url":"./treegridtest.dspy",

"idfield":"id",

"treefield":"name",

"fields":[

{"title":"name","field":"name"},

{"title":"gender","field":"gender"},

{"title":"age","field":"age"}

]

}

}