**组件使用文档（初）**

## 概述：

1高度定制的一些组件，

2注意：整个项目是基于jq库的

3 html 只需要一个div或者其他标签 无需css

4 javascript 只需要配置数据（如css样式数据等其他组件需要的数据）就可以实现样式等的定制

说明： options是个json对象的

每个控件都有默认的样式，也可以通过配置json对象中的css属性自己设置css样式

目前提供了9个组件

1组织目录

-----DemoHtml 组件demo文件夹

-----DemoTestJS 组件测试js文件

----img 默认样式图片资源文件夹

----js 组件源码文件夹

---data 数据文件夹（如全国省市区县地址）

--lib 支持库（jquery）

Indext.html 组件主页

README.md 一些注意事项和日志

详细组件：

1. 自动保存输入组件

描述：提供自动保存用户输入的信息组件

Js: AntoCompleteAndeSave.js

示例：

JavaScript调用： $("#inputTest").autoInput(options);

Html: <div>

<input id="inputTest" />

</div>

配置项：

Options：配置的json对象

css配置（见名知义）：

\* width:null, //

height:null, //线框的宽度 0 则表示没有

borderTopW:1,

borderBottomW:0,

borderRightW:0,

borderColor:"#ffffff",

bgColor:"rgba(256,256,256,0.8)",

mouseOverColor:"#EEB422", //鼠标进入的颜色

mouseOutColor:"#EEE5DE"

\*

\* 其他配置：

\* AutoSwitch:true, //开关

Input: null, //自定义输入变化函数

Propertychange:null, //自定义输入变化函数

Change:null, //自定义输入变化函数

MaxLine:5, //内容提示最大数目

1. 左右按钮组件

描述：左右按钮组件，可以生成

Js: LeftAndRightBtn.js

示例：

Javascript调用：$("#dd").leftAndRightBtn(); //默认

$("#dd").leftAndRightBtn(pageOptions);//配置项

Html: <div id="dd"/>

配置项：pageOptions

\* 1，配置点击 事件 ：leftBtnClick:function(parm){} 左键点击事件

\* rightBtnClick:function(parm){} 右键点击事件

\*

\* 2，css配置（见名知义）：

\* width: 120,

height:40,

radius:5,

backgroundColor:"rgba(3,254,239,1)",

boder:"solid",

borderColor:"rgba(3,254,239,0.8)",

borderWidth:0,

mouseOverColor:"rgba(0,0,0,0.8)",

clickColor:"#000000"

\*

\* 3，其他配置：

\* ImageWidth:20, 图片的大小

leftImagePath:"../img/trian\_left.png", 图片的路径

rightImagePath:"../img/trian\_right.png",

purePicBtn:false, 是否是纯图片按钮

imgGapWidth:120, 纯图片按钮是的间距 （可能这里效果要多调）

\*

1. 消息提示组件

描述：页面悬浮的消息提示组件，可在通过配置在页面顶部，中部，底部飘出提示，延迟消失,

Js: MessageComponent.js

示例：

Javascript调用：$("#yy").message(options0).show(); 调用show()方法可重复出现

Html: <div id="yy"/>

配置项：pageOptions

\* 1，配置点击 事件 ：clickClose:function(){} 点击关闭按钮

\*

\* 2，css配置（见名知义）：

\* height:30,

fontSize:16,

boder:1,

fontColor:"#ffffff",

bgColor:"rgba(3,254,239,1)",

contentColor:"rgba(3,254,239,0.8)",

btnWidth:30,

iconWidth:30,

radius:20,

imgWidth:10,

imgHeight:10

\*

\* 3，其他配置：

\* MessageType:"important", //common 普通消息, important 重要提示消 息

position:"top", ///消息提示框的位置， 上 top 中center 下 bottom

content:"这里是默认的提示！！",

btnImg:"../img/close.png",

1. 导航组件

描述：提供导航栏组件，根据配置的数据显示

Js: NavigationComponent.js

示例：

JavaScript调用： $("#ff").navigation(options);

Html: <div id =”ff”></div>

配置项：

Options：配置的json对象

1. css配置（见名知义）：

navWidth:180, //导航宽度

navHeight:120, //导航高度

bgColor:"rgba(0,0,0,0.8)", //导航背景色

mouseOverColor:"rgba(3,254,239,1)", //鼠标移的颜色

padingTop:15, //图片距离上面的距离

subNavWidht:300, // 子级导航宽度

subNavHeight:300, //子级导航高度

subNavLeft:200,

subNavTop:-120,

fontColor:"#ffffff",

subNavChildHeight:40,

subNavChildMarginTopGap:5,

iconWidth:30,

1. 数据（详见参考data文件夹的provice\_city\_area.js）

data: 一个json对象数组

//data是 json对象

//data是一个数组 是最外的导航栏

//data.subdata 是次级菜单数组

//data.subdata.subdata 是 次次级菜单数组

1. 分页组件

描述：分页组件，

Js: PageControlComponent.js

示例：

JavaScript调用：$("#wq").pageComponent(); //无任何配置 （简单的上下页）

$("#wq").pageComponent(options);

Html: <div id =”wq”></div>

配置项：

Options：配置的json对象

\* 1，配置点击 事件 ：pageItemClick:function(pageNum){}

\* pageNum为当前页index 若无 则0 表示上一页 -1 表示下一页

\* 2，css配置（见名知义）： circleWidth:60, 页码的宽度

circleHeight:60, 页码的高度

radius:30,

backgroundColor:"rgba(0,0,0,0)", 背景色

boder:"solid",

borderColor:"#000000",

borderWidth:1,

mouseOverColor:"rgba(0,0,0,0)",

mouseOverBoderColor:"#03feef",

mouseOverFontColor:"#03feef",

fontColor:"#FFFACD",

fontSize: 20,

gapWidth:20

\* 3，其他配置：

\* PageNum:0, 页总数

MaxPageNum:99, 最大页数

PrePageContent:"<<", 上一页的表现

LastPageContent:">>", 下一页的表现

CurrentPageNum:1, 当前页数

\*

1. 下拉菜单组件

描述：下拉菜单组件，提供无标题和有标题的多列（目前最多三列）灵活配置的下拉组件

Js: PullDownlComponent.js

示例：

JavaScript调用：

创建：var x = $("#ff").pullDown(options);

获取当前选择的值：

x. GetCurrentValue(); 返回时一个数组

Html: <div id =”ff”></div>

配置项：

Options：配置的json对象

css配置（见名知义）：

\* itemHeight:45, //一致的高度

boderRadius:5,

fontColor:"#ffffff",

titleWidth:100,

titleBgColor:"rgba(3,254,239,0.8)",

mouseOverColor:"rgba(0,0,0,0.5)",

selectWidth:200,

iconWidth:20,

selectDivHeight:200,

selectBgColor:'rgba(0,0,0,0.2)',

其他配置：

\* title:"地址", 这个是头部的题标 如果不设置就没有头部

pullDownIcon:"../img/up.png",

titleContent:[], 数组是控制多少个下拉菜单 数组值是默认的下 拉菜单显示值

data:[], data是一个json对象数组 格式如下（或者参照 data文件下的provice\_city\_area.js）

1. 搜索提示组件

描述：搜索提示组件，根据输入的内容，从后台获取相关提示

Js: SearchInput.js

示例：

JavaScript调用

创建：$("#rr").searchInput(options);

刷新提示： $("#rr").refrshInputComplete(arr);

arr 为后台请求提供的数组

Html: <div id =”rr”></div>

配置项：

Options：配置的json对象

\* 1，配置点击 事件 ：sreachBtnClick:function(str){} 点击搜索按 钮事件 str为 input的字符

\* itemBtnClick:function(i,content){} 点击了提示的itme， i为 index content item的value

\* inputChange:refreshStr

\* //输入框变化事件的回调函数 str 参数 为input的 value

\*

\* 2，css配置（见名知义）：

\* inputWidth:300,

inputHeight:40,

inputBorderRadius:5,

inputPaddingLeft:20,

inputPaddingRight:20,

inputFontSize:20,

inputFontColor:"#ffffff",

InputBgColor:"rgba(0, 0, 0, 0.5)",

imgDivWidth:null, //这里这个默认设为input的bgColor, 用户需要时自定义宽度

imgWidth:20,

imgDivBgColor:null, //这里这个默认设为input的bgColor, 用户需要时自定义按钮区域的bgColor

groupWidth:null, //和input一样的宽度

groupHeight:300,

groupTop:10,

groupBgColor:null, //这里这个默认设为null,用户需要 时自定义按钮区域的bgColor

itemWidth:145,

itemHeight:40,

itemBgColor:"#026C7A",

itemMarginTop: 20, //和上一行的间隔

itemMarginLeft:20,

itemFontSize: 15,

itemBoderRadius:5,

itemFontColor:"#ffffff",

itemBgColor: "#03feef",

opacity: 0.5

1. 提示组件

描述: 提示组件，鼠标移动上去显示提示，根据配置显示方向，内容等

Js: ToolTip.js

示例：

JavaScript调用：$("#e1").toolTip(options1);

Html: <div id =” e1”></div>

配置项：

Options：配置的json对象

css配置（见名知义）：

\* contentWidth:200,

contentHeight:100,

titleWidth: 200,

titleHeight: 40,

arrowWidth:30,

titleBgColor:"rgba(0,0,0,0.8)",

titleFontColor:"rgba(256,256,256,1)",

contentFontColor:"rgba(256,256,256,1)",

contentBgColor:"rgba(0,0,0,0.5)",

arrowBgColor:"rgba(0,0,0,0.8)",

boderRadius:5,

其他配置：

\* showDirection:"right", 显示方向

showTitle:false, 是否显示标题

title:"这里是默认标题",

content:"这里是默认内容这里是默认内容这里是默认内容这里是默 认内容这里是内容这里是内容",

1. 对话框组件

描述: 对话框组件

Js: DialogBoxComponent.js

示例：

JavaScript调用：$('#bb').dialogBox()

$('#bb').dialogBox(Options)

Html: <div id =” bb”></div>

配置项：

Options：配置的json对象

css配置（见名知义）：

left:0,

top:0,

width:0,

height:0,

bgColor:"rgba(0,0,0,0.5)",

boderColor:"rgba(3,254,239,1)",

boderRidius:5,

fontColor:"#ffffff",

其他配置：

\* //按钮配置 用个json数组配置显示文字和点击回调函数

\* buttonData:[

{text:"关闭",click:function(){console.log("点击关闭按钮");}},

{text:"取消",click:function(){console.log("点击取消按钮");}},

{text:"保存",click:function(){console.log("点击保存按钮");}}

],

title:"标题",

content:"内容内容。。。。。