## 什么是EasyUI

easyui是一种基于jQuery、Angular、Vue和React的用户界面插件集合。

## 2.EasyUI的优点

a.easyui为创建现代化，互动，JavaScript应用程序，提供必要的功能。

b.使用easyui你不需要写很多代码，你只需要通过编写一些简单HTML标记，就可以定义用户界面。

c.easyui是个完美支持HTML5网页的完整框架。

## 如何使用EasyUI

a.下载EasyUI

<https://www.jeasyui.net/download>

b.引入EasyUI

<link rel = "stylesheet" type = "text/css" href = "jquery-easyui-1.10.1/themes/default/easyui.css" >

<link rel = "stylesheet" type = "text/css" href = "jquery-easyui-1.10.1/themes/icon.css" >

<script type = "text/javascript" src = "jquery-easyui-1.10.1/jquery.min.js" ></script>

<script type = "text/javascript" src = "jquery-easyui-1.10.1/jquery.easyui.min.js" ></script>

## 对Jquery ajax针对Easyui进行封装

**知识储备**

1. 回调函数
2. $.extend()

**var** postAjax = **function**(url,successFunction,options){

**var** returnData;

**var** defaultOption = {

type : "post",

async : **false**,

contentType : 'application/json; charset=utf-8',

dataType : "json",

errorFunction : errorFunction,

successArguments:""

}

**var** currentOptions = $.extend(defaultOption,options);

$.ajax({

type : currentOptions.type,

url : url,

async : currentOptions.async,

contentType : currentOptions.contentType,

dataType : currentOptions.dataType,

beforeSend :**function**(){

$.messager.progress({title: '提示', text: '数据处理中，请稍候....'});

},

success : **function**(resultData) {

$.messager.progress('close');

**if** (resultData&&isSuccess(resultData)) {

**if** (currentOptions&&currentOptions.successArguments){

successFunction(resultData,currentOptions.successArguments);

returnData = resultData.data;

**return** returnData;

}**else**{

successFunction(resultData);

returnData = resultData.data;

**return** returnData;

}

} **else** {

**if** (resultData){

defaultOption.errorFunction(resultData);

} **else** {

internalError();

}

}

},

complete : **function**(){},

error : **function**(){

$.messager.progress('close');

internalError();

**return**;

}

});

**return** returnData;

}

**function** internalError() {

showErrorMsg(500 + " " + "程序内部错误")

}

**function** errorFunction(resultData){

$.messager.alert('提示', resultData.description, 'info')

}

**function** showErrorMsg(errorMsg) {

$.messager.alert('提示', errorMsg, 'error')

};

**function** showInfoMsg(errorMsg) {

$.messager.alert('提示', errorMsg, 'info')

};

**function** isSuccess(resultData){

**return** "10" == resultData.code;

}

## 5.布局(Layout)

### 1.特点:

a.布局（layout）是有五个区域（北区 north、南区 south、东区 east、西区 west 和中区 center）的容器。

b.中间的区域面板是必需的，边缘区域面板是可选的。

c.每个边缘区域面板可通过拖拽边框调整尺寸，也可以通过点击折叠触发器来折叠面板。

d.布局（layout）可以嵌套，因此用户可建立复杂的布局。

### 2.用法

a.通过标记创建布局（Layout）。

添加 'easyui-layout' class 到 <div></div> 标记。

b.使用JavaScript创建布局

$(selector).layout(options);

### 3.布局选项(Layout options)

fit：当设置为 true 时，就设置布局（layout）的尺寸适应它的父容器。

### 4.区域面板选项(Region Panel Options)

title:布局面板（layout panel）的标题文本。

region：定义布局面板（layout panel）的位置，

其值是下列之一：north、south、east、west、center。

### 5.实例

<div class="easyui-layout" data-options="fit:true">

<div data-options="region:'north'" style="height: 86px;overflow: hidden;">north</div>

<div data-options="region:'west'" title="模板导航" style="width: 210px;overflow: hidden;">west</div>

<div data-options="region:'center'">center</div>

<div data-options="region:'south'">south</div>

</div>

## 6.面板(Panel)

### 1.用法

a.通过标记创建面板(panel)

把 'easyui-panel' class 添加到 <div></div> 标记;

b.使用javaScript创建面板(panel)

$(selector).panel(options);

### 2.面板属性

1.id：面板（panel）的 id 属性。

2.title：显示在面板（panel）头部的标题文字。

3.width：设置面板（panel）的宽。

4.height:设置面板（panel）的高度。

5.fit:当设置为 true 时，面板（panel）的尺寸就适应它的父容器。

6.border:定义了是否显示面板（panel）的边框,默认true。

7.tools：自定义工具组，可能的值：

1、数组，每个元素包含 iconCls 和 handler 两个属性。

2、选择器，指示工具组。

## 7.选项卡(Tab)

### 1.用法

a.用法通过标签创建选项卡(Tab)

将'easyui-tabs' class 添加到 <div></div> 标记；

b.使用JavaScript创建选项卡(Tab)

$(selector).tabs(options);

### 2.属性

1.fit：当设置为 true 时，就设置标签页（Tabs）容器的尺寸以适应它的父容器,默认false。

2.border:当设置为 true 时，就显示标签页（Tabs）容器边框,默认true。

3.tools：放置在头部的左侧或右侧的工具栏，可能的值：

1、数组，指示工具组，每个工具选项都和链接按钮（Linkbutton）一样。

2、选择器，指示包含工具的 <div></div>。

### 3.事件

onContextMenu：当一个标签页面板（tab panel）被右键点击时触发

## 8.数据表格(datagrid)

### 1.用法

a.通过标签创建数据表格(datagrid)

将'easyui-datagrid' class添加到<table></table>标记;

b.使用 javascript 创建数据网格（datagrid）

<table class="dataGrid"></table>

$(".dataGrid").datagrid(options);

### 2.DataGrid属性（该属性扩展自面板（panel））

1.columns:数据网格（datagrid）的列（column）的配置对象;

2.fitColumns：设置为 true，则会自动扩大或缩小列的尺寸以适应网格的宽度并且防止水平滚动，默认false;

3.url：从远程站点请求数据的 URL;

4.data:要加载的数据;

5.singleSelect:设置为 true，则只允许选中一行，默认false;

6.pagination:设置为 true，则在数据网格（datagrid）底部显示分页工具栏,默认false。

7.idField:指示哪个字段是标识字段。

8.sortName:定义可排序的列。

9.sortOrder:定义列的排序顺序，只能用 'asc' 或 'desc',默认'asc'。

10.checkOnSelect：如果设置为 true，当用户点击某一行时，则会选中/取消选中复选框。

如果设置为 false 时，只有当用户点击了复选框时，才会选中/取消选中复选框。

11.selectOnCheck：如果设置为 true，点击复选框将会选中该行。

如果设置为 false，选中该行将不会选中复选框。

12.nowrap:设置为 true，则把数据显示在一行里。设置为 true 可提高加载性能,默认为true。

13.pageNumber:当设置了 pagination 属性时，初始化页码，默认1;

14.pageSize:当设置了 pagination 属性时，初始化页面尺寸,默认10;

15.toolbar:数据网格（datagrid）面板的头部工具栏。可能的值：

1、数组，每个工具选项与链接按钮（linkbutton）一样。

2、选择器，指向工具栏。

### 3.DataGrid事件（该属性扩展自面板（panel））

1.onLoadSuccess：当数据加载成功时触发

## 9.消息框(messager)

消息框（messager）提供不同样式的消息框，包括警示（alert）、确认（confirm）、提示（prompt）、进展（progress）等等。

所有的消息框都是异步的。用户可以在与消息框交互后使用回调函数来完成一些动作。

1.$.messager.alert('Warning','The warning message');

## 10.对话框(Dialog)

## 11.表单(form)

## 12.菜单(Menu)

## 13.按钮(messager)

## 14.树(Tree)

a.用法通过标签创建树(Tree)

将'**easyui-tree**' class 添加到 <ul></ul> 标记；

b.使用JavaScript创建选项卡(Tree)

$(selector).tree(options);