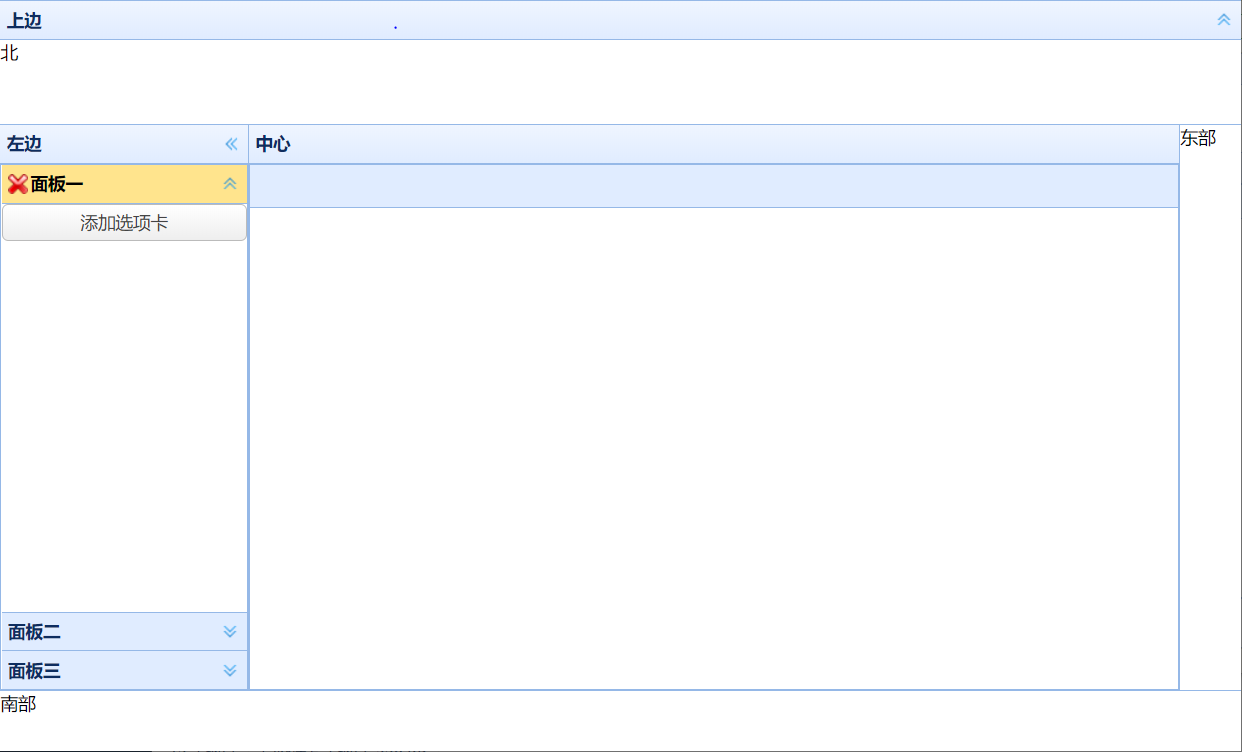
# 最简单布局步骤

## 效果图



## 第一步 导包

注意：要先引入jquery的js,再引入easyui的js和css。有先后顺序的。

<!-- 引入JQuery -->

<script type="text/javascript" src="${pageContext.request.contextPath} /jquery-easyui/jquery.min.js"></script>

<!-- 引入EasyUI -->

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

<!-- 引入EasyUI的中文国际化js，让EasyUI支持中文 -->

<script type="text/javascript" src=" /jquery-easyui/locale/easyui-lang-zh\_CN.js"></script>

<!-- 引入EasyUI的样式文件-->

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

<!-- 引入EasyUI的图标样式文件-->

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

## 第二步 布局

layout布局：north上、west左、east右、south下、center中心

在body标签 定义属性：class="easyui-layout"

在body标签体内加入div标签中有定义data-options属性

<!--使用div元素描述每个区域 region:定义区域的关键字，-->

<body class="easyui-layout">

<div title="上边" data-options="region:'north'" style="height:100px">北</div>

<div title="左边 " data-options="region:'west'" style="width:200px">西</div>

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

<div data-options="region:'east'" style="width:200px">东部</div>

<div data-options="region:'south'" style="height:50px">南部</div>

注意：需要定义样式style进行分离：高度上下和宽度左右，中心不用管，加了title就会有标题样式

## 第三步 定义折叠面板

定义折叠面板：在zuo标签div定义：accordion，标签内定义属性class="easyui-accordion"

可以添加属性fit:true 填充整个父容器 也可以添加：title、添加图标等

<!-- 在左面布局 折叠面板操作 -->

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

<div data-options="iconCls:'icon-cancel'" title="面板一">

<a id="show" class="easyui-linkbutton">查看 </a>

</div>

<div title="面板二">222</div>

<div title="面板三">333</div>

</div>

## 第四步 定义选项卡面板

选项卡面板 在center(中间)的div中放入如下内容

<div title="显示内容" data-options="region:'center'">

<div id="mytabs" class="easyui-tabs" data-options="fit:true">

<div data-options="iconCls:'icon-cancel',closable:true" title="选项一">111</div>

<div data-options="iconCls:'icon-cancel',closable:true" title="选项二">222</div>

<div data-options="iconCls:'icon-cancel',closable:true" title="选项三">333</div>

</div>

</div>

动态点击跳出选项卡操作

使用到 tabs的两个方法exists：判断是否已经打开选项卡，select：显示指定的选项卡

<script type="text/javascript">

$("#show").click(function(){

//第一步：通过按钮触发事件:如果标题有返回true，否则false

var e=$("#mytabs").tabs("exists","选项卡标题"); //此标签必须是属性带：class="easyui-tabs"

if(e){

//第二步 如果是true直接使用”select”方法打开此标题的窗口

$("#mytabs").tabs("select","选项卡标题");

}else{

第三步 ：没有此标题的方法，创建一个选项卡

$("#mytabs").tabs("add",{

title:" 选项卡标题",

closable:true, //标题中添加关闭按钮

content:”111”

})

}

})

</script>

# 选项卡

## HTMl配置

## JS动态调出

使用到方法有：

tabs("exists", "标题名");//有此标题名：返回true

tabs("select", "选项卡标题"); //指定标题名进行选中

tabs("add",{这里是json格式}; //添加选项卡，大括号有很多属性

使用iframe标签来显示选项卡中的内容

<iframe frameborder="0" width="100%" height="100%" src="index.html"></iframe>

iframe：老师说页面上挖出那么一个坑：可以单独显示独立页面

参数：frameborder="0"：边框为0，src：uri地址(1可以百度url，2相对资源文件)

<script>

//第一步： 通过按钮触发事件

$("#show").click(function() {

//第二步 使用exists方法打开是判断是否已经打开：标题为a的选项卡

var e = $("#mytabs").tabs("exists", "a");

if (e) {

//第三步 如果有则使用：使用select方法选中a选项卡

$("#mytabs").tabs("select", "a");

} else {

$("#mytabs").tabs("add", {

//第四步：没有标题为a的 使用add增加此标题a的选项卡

title: "a",

//标题中添加关闭按钮

closable: true,

//选项卡内容:这里引用iframe

content: '<iframe frameborder="0" width="100%" height="100%" src="index.html"></iframe>'

})

}

})

</script>

# Jquery插件Ztree的使用

简单的使用

第一步 定义页面那个位置使用Ztree

<ul id="ztree1" class="ztree"> //必须使用ul标签，属性必须有class="ztree"

</ul>

第二步 js部分

$(function(){

//第一步 属性的设置，不写 直接空就是默认的ztree属性

var setting={

data:{

simpleData:{

enable:true//使用简单的json数据构造ztree节点

}

}

};

//第二步 构造节点数据

var zNodes=[

//每个json对象表示一个节点,//使用了id代表自己的id，pId：代表父类的节点

{"id":"1","pId":"0","name":"节点一"},

{"id":"2","pId":"1","name":"节点二"},

{"id":"3","pId":"2","name":"节点三"}

];

//第三步 第一个参数：要使用的对象，setting设置的属性，节点数据

$.fn.zTree.init($("#ztree1"),setting,zNodes);

})

使用ztree实现动态点击跳出选项卡操作

$(function(){

//第一步 属性的设置，设置data变为简单json数据构造、绑定回调事件和点击事件

var setting={

data:{

simpleData:{

enable:true//使用简单的json数据构造ztree节点

}

},

callback:{ //ztree的回调事件

//为ztree节点绑定单机事件

onClick:function(event,treeId,treeNode){//treeNode其实就是json数据单节点全部内容

//alert("你点击了："+treeNode.name)

//判断选项卡是否已经存在

var e=$("#mytabs").tabs("exists",treeNode.name);

if(e){

$("#mytabs").tabs("select",treeNode.name);

}else{

//判断treeNode中有没有page。：因为设计的json数据只有叶子节点才有此数据

if(treeNode.page != undefined){

$("#mytabs").tabs("add",{

title:treeNode.name,

closable:true,

content:'<iframe frameborder="0" height="100%" width="100%" src="https://www.baidu.com"></iframe>' //跳转页面

});

}

}

}

}

};

//第二步 构造节点数据 动态使用json数据：ajax调用操作

var url ="${pageContext.request.contextPath}/js/menu.json";

$.get(url,{},function(data){

//第三步 参数：要使用的对象，setting设置的属性，节点数据

$.fn.zTree.init($("#ztree1"),setting,data);

},"json")

})

效果



# 用tree显示 tabs页面

$(function(){

$('#menu').tree({

onClick: function(node){

//叶子节点就是没有子节点的节点；子节点是相对于父节点来说的，它是父节点的下一层节点。

if($('#menu').tree("isLeaf",node.target)){//判断是不是叶子节点,是的话触发此事件，target参数是一个节点DOM对象。

var tabs = $("#tabs"); //首页的id

var tab = tabs.tabs("getTab",node.text);//选项卡 getTab 获取指定选项卡面板 node.text：节点名字

if(tab){ //判断是否有值，有值证明已经打开 :0\空字符\ false\ null\ undefined 的时候都是false

tabs.tabs("select",node.text);

}else{

tabs.tabs('add',{

title:node.text,//用叶子节点的名称命名

href: node.attributes.url, //让跳转属性

closable:true,//为true的是后 选项卡有关闭按钮

bodyCls:"content" //这个感觉有没有都可以

});

}

}

}

});

});