**框架地址：<http://www.layui.com/demo/>**

1. **初始化**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title></title>

<link rel="stylesheet" type="text/css" href="layui/css/layui.css"/>

<script src="js/jquery-2.1.0.js" type="text/javascript" charset="utf-8"></script>

<script src="layui/layui.js" type="text/javascript" charset="utf-8"></script>

</head>

<body>

</body>

</html>

1. **模块加载**

layui.use(['form', 'upload'], function(){ //如果只加载一个模块，可以不填数组。如：layui.use('form')

var form = layui.form(); //获取form模块

var upload = layui.upload; //获取upload模块

//使用模块

});

1. **模块示例**

**导航栏**

<ul class="layui-nav" lay-filter="demo">

<li class="layui-nav-item"><a href="">最新活动</a></li>

<li class="layui-nav-item layui-this"><a href="">产品</a></li>

<li class="layui-nav-item"><a href="">大数据</a></li>

<li class="layui-nav-item">

<a href="javascript:;">解决方案</a>

<dl class="layui-nav-child"> <!-- 二级菜单 -->

<dd><a href="">移动模块</a></dd>

<dd><a href="">后台模版</a></dd>

<dd><a href="">电商平台</a></dd>

</dl>

</li>

<li class="layui-nav-item"><a href="">社区</a></li>

</ul>

layui.use('element', function(){

var element = layui.element(); //导航的hover效果、二级菜单等功能，需要依赖element模块

//监听导航点击

element.on('nav(demo)', function(elem){

//console.log(elem)

layer.msg(elem.text());

});

});

**面包屑**

//默认

<span class="layui-breadcrumb">

<a href="/">首页</a>

<a href="/demo/">演示</a>

<a>导航元素</a>

</span>

//自定义分隔符

<span class="layui-breadcrumb" lay-separator="|">

<a href="">首页</a>

<a href="">国际新闻</a>

<a href="">亚太地区</a>

<a>正文</a>

</span>

**选项卡**

<div class="layui-tab" lay-filter="test">

<ul class="layui-tab-title">

<li class="layui-this">网站设置</li>

<li>用户管理</li>

<li>权限分配</li>

<li>商品管理</li>

<li>订单管理</li>

</ul>

<div class="layui-tab-content">

<div class="layui-tab-item layui-show">内容1</div>

<div class="layui-tab-item">内容2</div>

<div class="layui-tab-item">内容3</div>

<div class="layui-tab-item">内容4</div>

<div class="layui-tab-item">内容5</div>

</div>

</div>

**折叠面板**

在折叠面板的父容器设置属性 lay-accordion，始终只显示一个面板

<div class="layui-collapse">

<div class="layui-colla-item">

<h2 class="layui-colla-title">杜甫</h2>

<div class="layui-colla-content layui-show">内容区域</div>

</div>

<div class="layui-colla-item">

<h2 class="layui-colla-title">李清照</h2>

<div class="layui-colla-content">内容区域</div>

</div>

<div class="layui-colla-item">

<h2 class="layui-colla-title">鲁迅</h2>

<div class="layui-colla-content">内容区域</div>

</div>

</div>

**按钮**

<input type="button" class="layui-btn" id="lay\_alert" value="确定" />

<input type="button" class="layui-btn layui-btn-primary" id="lay\_alert" value="取消" />

<input type="button" class="layui-btn layui-btn-disabled" id="lay\_alert" value="按钮不可用" />

**表格**

|  |  |  |
| --- | --- | --- |
| **属性名** | **属性值** | **用途** |
| lay-even | 无 | 用于开启 偶数行 背景，可与其它属性一起使用 |
| lay-skin="属性值" | line/row/nob | line：定义行边框 风格表格  row：定义列边框 风格表格  nob：定义 无边框 风格表格 |

<div class="layui-form">

<table class="layui-table">

<thead>

<tr>

<th width="50"><input type="checkbox" name="" lay-skin="primary" lay-filter="allChoose"></th>

<th>人物</th>

<th>民族</th>

<th>出场时间</th>

<th>格言</th>

</tr>

</thead>

<tbody>

<tr>

<td><input type="checkbox" name="" lay-skin="primary"></td>

<td>贤心</td>

<td>汉族</td>

<td>1989-10-14</td>

<td>人生似修行</td>

</tr>

</tbody>

</table>

</div>

layui.use([‘form’], function(){

var form= layui.form();

//全选

form.on('checkbox(allChoose)', function(data){

var child = $(data.elem).parents('table').find('tbody input[type="checkbox"]');

child.each(function(index, item){

item.checked = data.elem.checked;

});

form.render('checkbox');

});

});

**表单集合**

|  |  |  |
| --- | --- | --- |
| **属性名** | **属性值** | **用途** |
| title | 任意字符 | 设定元素名称，一般用于checkbox、radio框 |
| lay-skin | switch（开关风格）  primary（原始风格） | 定义元素的风格，目前只对 checkbox 元素有效，可将其转变为开关样式 |
| lay-ignore | 任意字符或不设值 | 是否忽略元素美化处理。设置后，将不会对该元素进行初始化渲染，即保留系统风格 |
| lay-filter | 任意字符 | 事件过滤器，主要用于事件的精确匹配，跟选择器是比较类似的。其实它并不私属于Form模块，它在Layui的很多基于事件的接口中都会应用到。 |
| lay-verify | required（必填项） phone（手机号） email（邮箱） url（网址） number（数字） date（日期） identity（身份证） 自定义值 | 同时支持多条规则的验证，格式：lay-verify="验证A|验证B"  如：lay-verify="required|phone|number"  另外，除了我们内置的校验规则，你还可以给他设定任意的值，比如lay-verify="pass"，那么你就需要借助form.verify()方法对pass进行一个校验规则的定义 |

<form class="layui-form" action="">

<div class="layui-form-item">

<label class="layui-form-label">输入框</label>

<div class="layui-input-block layui-input-text">

<input type="text" name="title" required lay-verify="required" placeholder="请输入标题" autocomplete="off" class="layui-input">

</div>

</div>

<div class="layui-form-item">

<label class="layui-form-label">密码框</label>

<div class="layui-input-block layui-input-text">

<input type="password" name="password" required lay-verify="required" placeholder="请输入密码" autocomplete="off" class="layui-input">

</div>

</div>

<div class="layui-form-item">

<label class="layui-form-label">日期</label>

<div class="layui-input-inline">

<input type="text" name="date" required lay-verify="required" placeholder="请输入日期" autocomplete="off" class="layui-input" onclick="layui.laydate({elem: this, istime: true, format: 'YYYY-MM-DD hh:mm:ss'})">

</div>

</div>

<div class="layui-form-item">

<label class="layui-form-label">选择框</label>

<div class="layui-input-block" style="width: 200px ;">

<select name="city" lay-verify="required">

<option value=""></option>

<option value="0">北京</option>

<option value="1">上海</option>

<option value="2">广州</option>

<option value="3">深圳</option>

<option value="4">杭州</option>

</select>

</div>

</div>

<div class="layui-form-item">

<label class="layui-form-label">复选框</label>

<div class="layui-input-block">

<input type="checkbox" name="like[write]" title="write">

<input type="checkbox" name="like[read]" title="read" checked>

<input type="checkbox" name="like[dai]" title="dai" disabled="disabled">

</div>

</div>

<div class="layui-form-item">

<label class="layui-form-label">原始复选框</label>

<div class="layui-input-block">

<input type="checkbox" name="like[write]" title="write" lay-skin="primary">

<input type="checkbox" name="like[read]" title="read" checked lay-skin="primary">

<input type="checkbox" name="like[dai]" title="dai" lay-skin="primary" disabled="disabled">

</div>

</div>

<div class="layui-form-item">

<label class="layui-form-label">开关</label>

<div class="layui-input-block">

<input type="checkbox" name="switch" lay-skin="switch">

</div>

</div>

<div class="layui-form-item">

<label class="layui-form-label">单选框</label>

<div class="layui-input-block">

<input type="radio" name="n" value="1" title="选1">

<input type="radio" name="n" value="2" title="选2" checked>

<input type="radio" name="n" value="3" title="选3" disabled="disabled">

</div>

</div>

<div class="layui-form-item layui-form-text">

<label class="layui-form-label">文本域</label>

<div class="layui-input-block">

<textarea name="desc1" placeholder="请输入内容" class="layui-textarea"></textarea>

</div>

</div>

<div class="layui-form-item">

<div class="layui-input-block">

<button class="layui-btn" lay-submit lay-filter="formDemo">立即提交</button>

<button type="reset" class="layui-btn layui-btn-primary">重置</button>

</div>

</div>

</form>

layui.use(['form', 'laydate'], function(){

//日期弹框需要加载 laydate 模块

var form = layui.form();

//事件监听

form.on('event(lay-filter的值)', callback);

//例子

<http://www.layui.com/doc/modules/form.html>

//监听提交

form.on('submit(formDemo)', function(data){

layer.msg(JSON.stringify(data.field));

return false;

});

});

**地区联动**

<div class="layui-form-item layui-form-area">

<label class="layui-form-label">地区</label>

<div class="layui-input-block">

<span class="fieldSet">

<input type="hidden" id="areaId" name="memberAttribute\_4" treePath=",1,6" />

</span>

</div>

</div>

<script src="layui/lay/modules/jquert.lSelect.js"></script>

var $areaId = $("#areaId");

$areaId.lSelect({

url: "/area",

});

**消息弹框**

<http://layer.layui.com/>

layui.use(['layer'], function(){

//alert

layer.alert("弹框内容", {title: '弹框标题'});

//prompt

layer.prompt({title: '输入任何口令，并确认'}, function(text, index){

//确定

layer.close(index);

layer.msg(text);

});

//confirm

layer.confirm('消息内容', {title: 'confirm'}, function(){

//确定

}, function(){

//取消

})

//消息提示

layer.msg("显示消息");

//tips

$('#id').on('click', function(){

var that = this;

layer.tips('只想提示地精准些', that); //在元素的事件回调体中，直接赋予this即可

});

//自定义弹框

layer.open({

type: 1,

title: "自定义标题",

closeBtn: 1,

btn: ['按钮一', '按钮二'],

yes: function(index, layero){

//按钮【按钮一】的回调

layer.msg("按钮一")

},

btn2: function(index, layero){

//按钮【按钮二】的回调

layer.msg("按钮二")

//return false 开启该代码可禁止点击该按钮关闭

},

cancel: function(){

//右上角关闭回调

layer.msg("关闭")

//return false 开启该代码可禁止点击该按钮关闭

},

area: ['420px', '240px'], //宽高

content: "<div>几句话</div>" //自定义内容

});

**进度条**

<!--小-->

<div class="layui-progress" lay-showpercent="true">

<div class="layui-progress-bar" lay-percent="40%"></div>

</div>

<!--大-->

<div class="layui-progress layui-progress-big" lay-showpercent="true">

<div class="layui-progress-bar" lay-percent="40%"></div>

</div>

**图标**

<http://www.layui.com/doc/element/icon.html>

<i class="layui-icon" style="font-size: 30px; color: #F08B2F;">&#xe609;</i>

**分页**

|  |  |  |  |
| --- | --- | --- | --- |
| **键** | **默认值** | **类型** | **描述** |
| cont | 必填 | String/Object | 容器。值可以传入元素id或原生DOM或jquery对象，如：  code[layui.code](http://www.layui.com/doc/modules/code.html" \t "http://www.layui.com/doc/modules/_blank)  cont: 'id'  //cont: document.getElementById('id')  //cont: $('#id') |
| pages | 必填 | Number | 分页数。一般通过服务端返回得到 |
| curr | 1 | Number | 当前页。 |
| groups | 5 | Number | 连续分页数。 |
| skin | default | String | 控制分页皮肤。一般传16进制色值即可，如:  code[layui.code](http://www.layui.com/doc/modules/code.html" \t "http://www.layui.com/doc/modules/_blank)  skin: '#c00'  当然，您还可以借助自定义皮肤 |
| first | 1 | Number/String/Boolean | 用于控制首页。值支持三种类型。 如：first: '首页' 如：first: false，则表示不显示首页项 |
| last | 总页数值 | Number/String/Boolean | 用于控制尾页。值支持三种类型 如：last: '尾页' 如：last: false，则表示不显示尾页项 |
| prev | 上一页 | String/Boolean | 用于控制上一页。若不显示，设置false即可 |
| next | 下一页 | String/Boolean | 用于控制下一页。若不显示，设置false即可 |

<div id="demo1"></div>

// 分页

layui.use(['layer'], function(){

laypage({

cont: 'demo1' //容器

,pages: 100 //总页数

,groups: 5 //连续显示分页数

,skip: true //显示跳页

,last: '尾页' //显示尾页

,curr: location.hash.replace('#!page=', '') //获取hash值为page的当前页

,hash: 'page' //自定义hash值

});

});