

# LayUI

和 Bootstrap 有些相似，但该框架有个极大的好处就是定义了很多前后端交互的样式接口，如分页表格，只需在前端配置好接口，后端则按照定义好的接口规则返回数据，即可完成页面的展示，极大减少了后端人员的开发成本。

官网地址：<https://www.layui.com/>（已经停止维护）<https://www.layuion.com/>

项目下载地址：<https://github.com/sentsin/layui>，<https://gitee.com/sentsin/layui>

## 下载文件说明

官网：<http://layui-doc.pearadmin.com/>

可以从gitee直接下载项目，下载完成后解压，我们看以下其目录结构，并将其完整的拷贝到我们自己的项目下。

```
1 C:.\n2 |  gulpfile.js      // gulp文件配置\n3 |  package.json\n4 |  README.md\n5 |\n6 |─dist              // 编译好的文件目录\n7 | |  layui.js\n8 | |  └─css\n9 | | |  layui.css    // 编译好的css\n10 | | |  └─modules    // 编译好的模块目录\n11 | | |      └─laydate\n12 | | |      └─layer\n13 |─doc               // 在线文档地址\n14 |─examples          // 案例目录\n15 |─src\n16 | |  layui.js\n17 | |  └─css\n18 | | |  layui.css    // 样式文件\n19 | | |  └─modules    // 内置模块的样式文件目录\n20 | |─font            // 字体文件目录\n21 | |─modules         // 内置模块目录\n22 | |      all.js\n23 | |      carousel.js\n24 | |      code.js\n25 | |      colorpicker.js\n26 | |      demo.js\n27 | |      dropdown.js\n28 | |      element.js\n29 | |      flow.js\n30 | |      form.js\n31 | |      jquery.js\n32 | |      lay.js\n33 | |      laydate.js\n34 | |      layedit.js\n35 | |      layer.js\n36 | |      laypage.js
```

```

37 |         laytpl.js
38 |         layui.all.js
39 |         mobile.js
40 |         rate.js
41 |         slider.js
42 |         table.js
43 |         transfer.js
44 |         tree.js
45 |         upload.js
46 |         util.js
47 |
48 | └─test                // 测试文件目录

```

css 样式

font:字体

image:图片

dist: 编译好的模块文件

使用时我们只需引入下述两个文件即可使用

```

1 | <!-- LayUI的核心CSS文件 -->
2 | <link rel="stylesheet" href="static/css/layui.css">
3 | <!-- LayUI的核心JS文件（采用模块化引入） -->
4 | <script src="static/layui.js"></script>

```

layui.js 框架 如果要使用必须使用

```

1 | layui.use(["table", "layer", "laydate", "form"], function() {});

```

## 使用方法

创建项目并引入layui

```

1 | <!DOCTYPE html>
2 | <html lang="en">
3 | <head>
4 |     <meta charset="UTF-8">
5 |     <title>Title</title>
6 |     <!-- 导入 css 样式-->
7 |     <link rel="stylesheet" href="static/css/layui.css">
8 | </head>
9 | <body>
10 | <!-- 你的 HTML 代码 -->
11 | <!--引入JavaScript代码-->
12 | <script src="static/layui.js"></script>

```

```
13 <script>
14   layui.use(['form'], function () {
15     let form = layui.form
16   })
17 </script>
18 </body>
19 </html>
```

## 布局

### 布局容器

#### 固定宽度

- 将栅格放入一个带有 class="layui-container" 的特定的容器中，以便在小屏幕以上的设备中固定宽度，让列可控。
- 固定宽度的两侧有留白效果

```
1 <div class="layui-container" style="background-color: pink;height: 300px;">
2     固定宽度（两侧有留白效果）
3 </div>
```

#### 完整宽度

- 可以不固定容器宽度。将栅格或其它元素放入一个带有 class="layui-fluid"的容器中，那么宽度将不会固定，而是 100% 适应
- 完整宽度占据屏幕宽度的100%

```
1 <div class="layui-fluid" style="background-color: cyan;height: 300px;">
2     完整宽度(占据屏幕宽度的100%)
3 </div>
```

### 栅格系统

我们将容器进行了 12 等分，预设了 4\*12 种 CSS 排列类，它们在移动设备、平板、桌面中/大尺寸四种不同的屏幕下发挥着各自的作用。

#### 栅格布局规则

- 采用 class="layui-row" 来定义行，如：

```
1 <div class="layui-row"></div>
```

- 采用类似 layui-col-md\* 这样的预设类来定义一组列(column)，且放在行(row)中
  - 变量 md 代表的是不同屏幕下的标记
  - 变量 \* 代表的是该列所占用的12等分数（如6/12），可选值为 1 - 12
  - 如果多个列的“等分数值”总和等于12，则刚好满行排列。如果大于12，多余的列将自动另起一行。
- 列可以同时出现最多四种不同的组合，分别是：xs（超小屏幕，如手机）、sm（小屏幕，如平板）、md（桌面中等屏幕）、lg（桌面大型屏幕），以呈现更加动态灵活的布局。
- 可对列追加类似 layui-col-space5、layui-col-md-offset3 这样的预设类来定义列的间距和偏移。
- 可以在列（column）元素中放入你自己的任意元素填充内容，完成布局！

```

1  <!--
2      栅格系统
3      列组合
4          1. 定义行    .layui-row
5          2. 定义列    .layui-col-md*
6              md 表示不同屏幕的标识（xs、sm、md、lg）
7              * 表示列的数量
8          3. 每一行被均分为12列，列的总数不能超过12，否则会自动换行
9      -->
10
11 <!-- 布局容器 -->
12 <div class="layui-container">
13     <!-- 定义行 -->
14     <div class="layui-row">
15         <!-- 定义列 -->
16         <div class="layui-col-md5" style="background-color: deepskyblue;">
17             内容5/12
18         </div>
19         <div class="layui-col-md7" style="background-color: bisque;">
20             内容7/12
21         </div>
22     </div>
23
24
25     <!-- 定义行 -->
26     <div class="layui-row">
27         <!-- 定义列 -->
28         <div class="layui-col-md4" style="background-color: powderblue;">
29             内容4/12
30         </div>
31         <div class="layui-col-md4" style="background-color:
32 mediumaquamarine;">
33             内容4/12
34         </div>
35         <div class="layui-col-md6" style="background-color: grey;">
36             内容6/12
37         </div>
38     </div>
39 </div>

```

## 响应式规则

- 简单来说，就是会针对四类不同尺寸的屏幕，进行响应式适配处理。

```

1  <!--
2      4. 响应式规则
3      栅格会自动根据屏幕的分辨率选择对应的样式效果。
4  -->
5
6  <body>
7      <h3>平板、桌面端的不同表现:</h3>
8      <div class="layui-row">
9          <!-- 小屏幕占6列，中屏幕占4列 -->
10         <div class="layui-col-sm6 layui-col-md4" style="background-color:
11 thistle">
12             平板≥768px: 6/12 | 桌面端≥992px: 4/12

```

```

12     </div>
13 </div>
14
15 <div class="layui-row">
16     <!-- 小屏幕占4列，中屏幕占6列 -->
17     <div class="layui-col-sm4 layui-col-md6" style="background-color:
18     mediumaquamarine;">
19         平板≥768px: 4/12 | 桌面端≥992px: 6/12
20     </div>
21 </div>
22
23 <div class="layui-row">
24     <!-- 小屏幕占12列，中屏幕占8列 -->
25     <div class="layui-col-sm12 layui-col-md8" style="background-color:
26     coral">
27         平板≥768px: 12/12 | 桌面端≥992px: 8/12
28     </div>
29 </div>
30 </body>

```

## 列间距

- 设定列之间的间距
- 且一行中最左的列不会出现左边距，最右的列不会出现右边距
- 列间距在保证排版美观的同时，还可以进一步保证分列的宽度精细程度。
- 给容器添加class="layui-col-space\*"
  - 支持1px-30px区间的所有双数间隔，以及 1px、5px、15px、25px 的单数间隔

```

1 <!-- 列边距 .layui-col-space* -->
2 <h3>列边距</h3>
3 <div class="layui-row layui-col-space20">
4     <div class="layui-col-md4">
5         <!-- 给具体的内容设置背景颜色 -->
6         <div style="background-color: hotpink">4</div>
7     </div>
8     <div class="layui-col-md4">
9         <div style="background-color: indianred">4</div>
10    </div>
11    <div class="layui-col-md4">
12        <div style="background-color: powderblue">4</div>
13    </div>
14 </div>
15

```

注意：

- layui-col-space 设置后不起作用主要是因为设置的是 padding，也就是说向内缩，所以设置背景色 padding 也是会添上颜色，看起来好像没有间距一样。可以在里面加一个 div 来达到目的
- 如果需要的间距高于30px（一般不常见），我们需要采用列偏移

## 列偏移

- 对列追加类似 layui-col-md-offset\* 的预设类，从而让列向右偏移
  - 其中 \* 号代表的是偏移占据的列数，可选中为 1 - 12
  - 如 layui-col-md-offset3，即代表在 中型桌面屏幕下，让该列向右偏移3个列宽度。

```
1 <body>
2   <h3>列偏移</h3>
3   <div class="layui-row">
4     <div class="layui-col-md4">
5       <div style="background-color: red">4</div>
6     </div>
7
8     <!-- 向右移动4列 -->
9     <div class="layui-col-md4 layui-col-md-offset4">
10      <div style="background-color: skyblue">
11        向右移动4列
12      </div>
13    </div>
14  </div>
15 </body>
16
```

## 列嵌套

- 列之间可以无限嵌套列

```
1 <div class="layui-row">
2   <!-- 大的盒子占6列 -->
3   <div class="layui-col-md6">
4     <div style="background-color: red">
5       <div class="layui-row">
6         <!-- 嵌套列 -->
7         <div class="layui-col-md3" style="background-color:
8           burlywood;">
9           内部列
10        </div>
11        <div class="layui-col-md5" style="background-color:
12          indianred;">
13          内部列
14          <div class="layui-col-md2" style="background-color:
15            mediumaquamarine;">
16              内部列
17            </div>
18          </div>
19        </div>
20      </div>
21    </div>
22  </div>
```

# 按钮

## 按钮风格

- 向任意 HTML 元素设定 class="layui-btn"，建立一个基础按钮。
- 通过追加样式为 class="layui-btn-{type}" 来定义其他按钮风格

名称	组合
原始	class="layui-btn layui-btn-primary"
默认	class="layui-btn"
百搭	class="layui-btn layui-btn-normal"
暖色	class="layui-btn layui-btn-warm"
警告	class="layui-btn layui-btn-danger"
禁用	class="layui-btn layui-btn-disabled"

```
1 <div class="layui-container">
2   <!-- 基础按钮 -->
3   <button type="button" class="layui-btn">一个标准的按钮</button>
4   <a href="http://www.layui.com" class="layui-btn">一个可跳转的按钮</a>
5   <div class="layui-btn">一个按钮</div>
6   <hr>      <!-- 不同主题的按钮 -->
7   <button class="layui-btn">默认按钮</button>
8   <button class="layui-btn layui-btn-primary">原始按钮</button>
9   <button class="layui-btn layui-btn-normal">百搭按钮</button>
10  <button class="layui-btn layui-btn-warm">暖色按钮</button>
11  <button class="layui-btn layui-btn-danger">警告按钮</button>
12  <button class="layui-btn layui-btn-disabled">禁用按钮</button>
13 </div>
14
15 <script src="static/layui.js"></script>
16 <script>
17   layui.use(function () {
18     let layer = layui.layer
19     let $ = layui.$
20   })
21 </script>
22 </body>
23
```

## 按钮尺寸

尺寸	组合
大型	class="layui-btn layui-btn-lg"
默认	class="layui-btn"
小型	class="layui-btn layui-btn-sm"
迷你	class="layui-btn layui-btn-xs"
流体按钮(最大化适应)	class="layui-btn layui-btn-fluid"

```
1 <div class="layui-container">
2   <!-- 不同尺寸的按钮 -->
3   <button class="layui-btn layui-btn-primary layui-btn-lg">大型原始按钮
4   </button>
5   <button class="layui-btn">默认按钮</button>
6   <button class="layui-btn layui-btn-sm layui-btn-danger">小型警告按钮
7   </button>
8   <button class="layui-btn layui-btn-xs">迷你按钮</button>
9   <button type="button" class="layui-btn layui-btn-fluid">流体按钮（最大化适
  应）</button>
10 </div>
```

## 圆角按钮

主题	组合
原始	class="layui-btn layui-btn-radius layui-btn-primary"s
默认	class="layui-btn layui-btn-radius"
百搭	class="layui-btn layui-btn-radius layui-btn-normal"
暖色	class="layui-btn layui-btn-radius layui-btn-warm"
警告	class="layui-btn layui-btn-radius layui-btn-danger"
禁用	class="layui-btn layui-btn-radius layui-btn-disabled"

```
1 <div class="layui-container" style="margin-top: 20px;"> <!-- layui-btn-
2   radius 圆角按钮 -->
3   <button class="layui-btn layui-btn-radius">默认圆角按钮</button>
4   <button class="layui-btn layui-btn-primary layui-btn-radius">原始圆角按钮
5   </button>
6   <button class="layui-btn layui-btn-normal layui-btn-radius">百搭圆角按钮
7   </button>
8   <button class="layui-btn layui-btn-warm layui-btn-radius">暖色圆角按钮
9   </button>
10  <button class="layui-btn layui-btn-danger layui-btn-radius">警告圆角按钮
11  </button>
12  <button class="layui-btn layui-btn-disabled layui-btn-radius">禁用圆角按钮
13  </button>
14 </div>
```



## 图标按钮

### 图标

- 对 i 标签 设定 class="layui-icon"
- 然后对元素加上图标对应的 font-class
- 内置图标一览表: <https://www.layui.com/doc/element/icon.html>

```
1 <i class="layui-icon layui-icon-face-smile"></i>
2
```

### 图标按钮

```
1 <div class="layui-container" style="margin-top: 20px;">
2   <!-- 图标按钮 -->
3   <button type="button" class="layui-btn"><i class="layui-icon">#xe608;
4   </i> 添加</button>
5   <button type="button" class="layui-btn layui-btn-sm layui-btn-primary">
6   <i class="layui-icon">#x1002;</i> 刷新
7   </button>
8   <button type="button" class="layui-btn layui-btn-sm layui-btn-warm"><i
9   class="layui-icon layui-icon-heart"></i> 关注
10  </button>
11 </div>
```

## 导航

- 导航一般应用于头部和侧边，是整个网页画龙点睛般的存在
- 依赖加载模块：element

实现步骤：

#### 1. 引入的资源

```
1 <link rel="stylesheet" href="static/css/layui.css">
2
```

#### 2. 依赖加载模块

```
1 <script>
2   // 导航 依赖 element 模块, 否则无法进行功能性操作
3   layui.use('element', function () {
4     var element = layui.element
5   })
6 </script>
7
```

## 水平导航

- 给一个无序列表 ul 添加 class="layui-nav"
- 给 li 添加 class="layui-nav-item" 表示的是导航的子项
- 给 li 添加 class="layui-this" 表示当前被选中的项
- 给 li 里面容器添加 class="layui-nav-child" 表示的是二级菜单

```
1  <!--  
2      水平导航      layui-nav  
3          layui-nav-item 表示的是导航的子项  
4          layui-this 表示当前被选中的项  
5          layui-nav-child 表示的是二级菜单  
6  -->  
7  <!-- 水平导航 layui-nav -->  
8  <ul class="layui-nav">  
9      <li class="layui-nav-item"><a href="">最新活动</a></li>  
10     <li class="layui-nav-item layui-this"><a href="">产品</a></li>  
11     <li class="layui-nav-item"><a href="">大数据</a></li>  
12     <li class="layui-nav-item">  
13         <a href="">解决方案</a>  
14         <!--二级菜单-->  
15         <dl class="layui-nav-child">  
16             <dd><a href="">移动模块</a></dd>  
17             <dd><a href="">后台模块</a></dd>  
18             <dd><a href="">电商平台</a></dd>  
19         </dl>  
20     </li>  
21 </ul>  
22  
23
```

## 图片与徽章

除了一般的文字导航，我们还内置了图片和徽章的支持，如：

```
1  <!--  
2      水平导航      layui-nav  
3          layui-nav-item 表示的是导航的子项  
4          layui-this 表示当前被选中的项  
5          layui-nav-child 表示的是二级菜单  
6  -->  
7  
8  <!-- 水平导航 -->  
9  <ul class="layui-nav" style="margin-top: 5px;">  
10     <!-- 导航的子项 -->  
11     <li class="layui-nav-item">  
12  
13         <a href="">控制台<span class="layui-badge">9</span></a>  
14     </li>  
15     <!-- 导航的子项 -->  
16     <li class="layui-nav-item">  
17         <a href="">个人中心<span class="layui-badge-dot"></span></a>  
18     </li>  
19     <!-- 导航的子项 -->  
20     <li class="layui-nav-item">  
21         <a href="">我</a>
```

```

22         <dl class="layui-nav-child">
23             <dd><a href="#">修改信息</a></dd>
24             <dd><a href="#">安全管理</a></dd>
25             <dd><a href="#">退出</a></dd>
26         </dl>
27     </li>
28 </ul>
29

```

## 导航主题

通过对导航追加CSS背景类，让导航呈现不同的主题色

- 给无序列表 ul 添加 class="layui-nav layui-bg-green" 可设置墨绿色背景的导航
- 水平导航支持其他的背景主题有：layui-bg-cyan(藏青)、layui-bg-molv(墨绿)、layui-bg-blue(艳蓝)
- 垂直导航支持的其他背景颜色有：layui-bg-cyan (藏青)

```

1  <!-- 墨绿 -->
2  <ul class="layui-nav layui-bg-green">
3      <li class="layui-nav-item"><a href="">最新活动</a></li>
4      <li class="layui-nav-item layui-this"><a href="">产品</a></li>
5      <li class="layui-nav-item"><a href="">大数据</a></li>
6      <li class="layui-nav-item">
7          <a href="">解决方案</a>
8          <!-- 二级菜单 -->
9          <dl class="layui-nav-child">
10             <dd><a href="">移动模块</a></dd>
11             <dd><a href="">后台模块</a></dd>
12             <dd><a href="">电商平台</a></dd>
13         </dl>
14     </li>
15 </ul>
16
17 <hr>
18 <!-- 藏青 -->
19 <ul class="layui-nav layui-bg-cyan">
20     <li class="layui-nav-item"><a href="">最新活动</a></li>
21     <li class="layui-nav-item layui-this"><a href="">产品</a></li>
22     <li class="layui-nav-item"><a href="">大数据</a></li>
23     <li class="layui-nav-item">
24         <a href="">解决方案</a>
25         <!-- 二级菜单 -->
26         <dl class="layui-nav-child">
27             <dd><a href="">移动模块</a></dd>
28             <dd><a href="">后台模块</a></dd>
29             <dd><a href="">电商平台</a></dd>
30         </dl>
31     </li>
32 </ul>
33
34 <!-- 艳蓝 -->
35 <hr>
36 <ul class="layui-nav layui-bg-blue">
37     <li class="layui-nav-item"><a href="">最新活动</a></li>

```

```

38     <li class="layui-nav-item layui-this"><a href="">产品</a></li>
39     <li class="layui-nav-item"><a href="">大数据</a></li>
40     <li class="layui-nav-item">
41         <a href="">解决方案</a>
42         <!-- 二级菜单 -->
43         <dl class="layui-nav-child">
44             <dd><a href="">移动模块</a></dd>
45             <dd><a href="">后台模块</a></dd>
46             <dd><a href="">电商平台</a></dd>
47         </dl>
48     </li>
49 </ul>
50

```

## 垂直导航

- 给无序列表 ul 添加 class="layui-nav layui-nav-tree"
- 给 li 添加 class="layui-nav-item layui-nav-itemed" 表示此子项目是默认展开的

```

1  <!-- 垂直导航 layui-nav layui-nav-tree -->
2  <ul class="layui-nav layui-nav-tree">
3      <li class="layui-nav-item layui-nav-itemed">
4          <a href="#">默认展开</a>
5          <dl class="layui-nav-child">
6              <dd><a href="#">选项1</a></dd>
7              <dd><a href="#">选项2</a></dd>
8              <dd><a href="">跳转</a></dd>
9          </dl>
10     </li>
11     <li class="layui-nav-item layui-nav-itemed">
12         <a href="#">解决方案</a>
13         <dl class="layui-nav-child">
14             <dd><a href="">移动模块</a></dd>
15             <dd><a href="">后台模版</a></dd>
16             <dd><a href="">电商平台</a></dd>
17         </dl>
18     </li>
19     <li class="layui-nav-item"><a href="">产品</a></li>
20     <li class="layui-nav-item"><a href="">大数据</a></li>
21 </ul>
22

```

## 侧边导航

- 给无序列表 ul 添加 class="layui-nav layui-nav-tree layui-nav-side"
- 设定layui-this来指向当前页面分类。

```

1  <!-- 侧边导航: <ul class="layui-nav layui-nav-tree layui-nav-side"> -->
2  <ul class="layui-nav layui-nav-tree layui-nav-side">
3      <li class="layui-nav-item layui-nav-itemed"><a href="#">默认展开</a>
4          <dl class="layui-nav-child">
5              <dd><a href="#">选项1</a></dd>
6              <dd><a href="#">选项2</a></dd>
7              <dd><a href="">跳转</a></dd>
8          </dl>
9      </li>

```

```

10     <li class="layui-nav-item layui-nav-itemed"><a href="#">解决方案</a>
11         <dl class="layui-nav-child">
12             <dd><a href="#">移动模块</a></dd>
13             <dd><a href="#">后台模版</a></dd>
14             <dd><a href="#">电商平台</a></dd>
15         </dl>
16     </li>
17     <li class="layui-nav-item layui-this"><a href="#">产品</a></li>
18     <li class="layui-nav-item"><a href="#">大数据</a></li>
19 </ul>
20

```

## 总结

水平、垂直、侧边三个导航的 HTML 结构是完全一样的，不同的是

- 水平导航：class="layui-nav"
- 垂直导航需要追加：class="layui-nav-tree"
- 侧边导航需要追加：class="layui-nav-tree layui-nav-side"

## 面包屑导航

- 给 span 标签 添加 layui-breadcrumb

```

1 <span class="layui-breadcrumb">
2     <a href="#">首页</a>
3     <a href="#">国际新闻</a>
4     <a href="#">亚太地区</a>
5     <a><cite>正文</cite></a>
6 </span>
7

```

- 我们还可以通过设置属性 lay-separator="-" 来自定义分隔符

```

1 <div class="layui-container">
2     <!-- 面包屑式导航 -->
3     <span class="layui-breadcrumb">
4         <a href="#">首页</a>
5         <a href="#">国际新闻</a>
6         <a href="#">亚太地区</a>
7         <a><cite>正文</cite></a>
8     </span>
9     <hr>
10    <!-- 设置属性 lay-separator="" 来自定义分隔符 -->
11    <span class="layui-breadcrumb" lay-separator="-">
12        <a href="#">首页</a>
13        <a href="#">国际新闻</a>
14        <a href="#">亚太地区</a>
15        <a><cite>正文</cite></a>
16    </span>
17    <hr>
18    <span class="layui-breadcrumb" lay-separator="|">
19        <a href="#">娱乐</a>
20        <a href="#">八卦</a>
21        <a href="#">体育</a>
22        <a href="#">搞笑</a>

```

```

23     <a href="">视频</a>
24     <a href="">游戏</a>
25     <a href="">综艺</a>
26 </span>
27 </div>
28
29

```

## 选项卡

- 导航菜单可应用于头部和侧边，Tab选项卡提供多套风格，支持响应式，支持删除选项卡等功能。
- 依赖加载模块：element

实现步骤：

### 1. 引入的资源

```

1 <link rel="stylesheet" href="static/css/layui.css">
2

```

### 2. 依赖加载模块

```

1 <script src="static/layui.js"></script>
2 <script>
3     // 注意：选项卡 依赖 element 模块，否则无法进行功能性操作
4     layui.use('element', function(){
5         var element = layui.element;
6
7     });
8 </script>
9

```

## 选项卡风格

- 默认风格给容器添加：class="layui-tab"
- 简洁风格给容器追加：class="layui-tab-brief"
- 卡片风格给容器需要追加：class="layui-tab-card"

```

1 <body>
2 <!-- 默认风格 .layui-tab -->
3 <div class="layui-tab">
4     <!-- 设置选项卡标题 .layui-tab-title -->
5     <ul class="layui-tab-title">
6         <li>网站设置</li>
7         <li>用户管理</li>
8         <li class="layui-this">权限分配</li>
9         <li>商品管理</li>
10        <li>订单管理</li>
11    </ul>
12    <!-- 设置选项卡的内容 .layui-tab-content -->
13    <div class="layui-tab-content">
14        <div class="layui-tab-item">内容1</div>
15        <div class="layui-tab-item">内容2</div>
16        <!-- 默认显示此内容 -->
17        <div class="layui-tab-item layui-show">内容3</div>

```

```

18         <div class="layui-tab-item">内容4</div>
19         <div class="layui-tab-item">内容5</div>
20     </div>
21 </div>
22 <hr>
23 <!-- 简洁风格 -->
24 <div class="layui-tab layui-tab-brief">
25     <ul class="layui-tab-title">
26         <li class="layui-this">网站设置</li>
27         <li>用户管理</li>
28         <li>权限分配</li>
29         <li>商品管理</li>
30         <li>订单管理</li>
31     </ul>
32     <div class="layui-tab-content">
33         <div class="layui-tab-item layui-show">内容1</div>
34         <div class="layui-tab-item">内容2</div>
35         <div class="layui-tab-item">内容3</div>
36         <div class="layui-tab-item">内容4</div>
37         <div class="layui-tab-item">内容5</div>
38     </div>
39 </div>
40 <hr>
41 <!-- 卡片风格 -->
42 <div class="layui-tab layui-tab-card">
43     <ul class="layui-tab-title">
44         <li class="layui-this">网站设置</li>
45         <li>用户管理</li>
46         <li>权限分配</li>
47         <li>商品管理</li>
48         <li>订单管理</li>
49     </ul>
50     <div class="layui-tab-content">
51         <div class="layui-tab-item layui-show">内容1</div>
52         <div class="layui-tab-item">内容2</div>
53         <div class="layui-tab-item">内容3</div>
54         <div class="layui-tab-item">内容4</div>
55         <div class="layui-tab-item">内容5</div>
56     </div>
57 </div>
58 <script src="static/layui.js"></script>
59 <script>
60     layui.use(function () {
61         let element = layui.element
62     })
63 </script>
64 </body>
65

```

## 表格

### 常规用法

1. 给 table 标签增加 class="layui-table"
2. 在colgroup 标签中定义表格列的宽度

- o `<col width="数值">` 表示相应列所占的宽度

### 3. `thead` 标签表示表格的头部区域

- o `tr` 标签用于定义表格中的行(行头)
  - `th` 标签用于定义表格中的表头(列头)

```
1 <body>
2 <!--
3     表格
4     class="layui-table"
5     常用属性
6     lay-even 如果设置了该属性，则可以显示隔行换色的效果
7     lay-skin 设置表格边框风格
8         line (行边框风格)
9         row (列边框风格)
10        nob (无边框风格)
11    lay-size 设置表格的尺寸
12        sm (小尺寸)
13        lg (大尺寸)
14 -->
15
16 <!-- 基础表格 .layui-table -->
17 <table class="layui-table">
18     <colgroup>
19         <!-- 第一列宽 150px 第二列宽300px 第三列宽自适应100% -->
20         <col width="150">
21         <col width="300">
22         <col>
23     </colgroup>
24     <!-- thead 标签表示表格的头部区域,其内部必须拥有 tr 标签，tr标签一般位于第一行 --
25     <thead>
26         <!-- tr用于定义表格中的行,必须嵌套在 table 中 -->
27         <tr>
28             <!-- th 用于定义表格中的表头,必须嵌套在 tr 中 -->
29             <th>昵称</th>
30             <th>加入时间</th>
31             <th>签名</th>
32         </tr>
33     </thead>
34
35     <tbody>
36     <tr>
37         <td>贤心</td>
38         <td>2016-11-29</td>
39         <td>人生就像是一场修行</td>
40     </tr>
41     <tr>
42         <td>许闲心</td>
43         <td>2016-11-28</td>
44         <td>于千万人之中遇见你所遇见的人，于千万年之中，时间的无涯的荒野里...</td>
45     </tr>
46     </tbody>
47 </table>
48
49 <script src="static/layui.js"></script>
50 <script type="text/javascript">
```



```

51     layui.use(['element', 'jquery'], function () {
52         var $ = layui.jquery
53     })
54 </script>
55 </body>
56

```

## 基础属性

静态表格支持以下基础属性，可定义不同风格/尺寸的表格样式：

属性名	属性值	备注
lay-even	无	用于开启隔行背景，可与其他属性一起使用
lay-skin="属性值"	line(行边框风格) row(列边框风格) nob(无边框风格)	若使用默认风格不设置该属性即可
lay-size="属性值"	sm(小尺寸) lg(大尺寸)	若使用默认尺寸不设置该属性即可

将我们所需要的基础属性写在 table 标签上即可：如（一个带有隔行背景，且行边框风格的大尺寸表格）：

```

1 <table lay-even lay-skin="line" lay-size="lg">
2
3 </table>
4

```

## 表单

- 依赖加载模块：form

实现步骤：

### 1. 引入的资源

```

1 <link rel="stylesheet" href="static/css/layui.css">
2

```

### 2. 依赖加载模块

```

1 <!-- 加载模块 -->
2 <script type="text/javascript">
3     // 加载form模块
4     layui.use("form",function(){
5         var form = layui.form;
6     });
7 </script>
8

```

### 3. 在一个容器中设定 class="layui-form" 来标识一个表单元素块

```

1 <form class="layui-form" action="">
2
3 </form>
4

```

#### 4. 基本的行区块结构，它提供了响应式的支持。

- 给 div 添加 class="layui-form-item" 代表“行”
- 给 label 添加 class="layui-form-label" 代表“区”
- 给 div 添加 class="layui-input-inline" 代表“块”

```

1 <body>
2
3 <!-- 在一个容器中设定 class="layui-form" 来标识一个表单元素块 -->
4 <form action="" class="layui-form">
5     <!-- 基本的行区块结构，它提供了响应式的支持。-->
6     <div class="layui-form-item">
7         <label class="layui-form-label">标题区域</label>
8         <div class="layui-input-inline">
9             <!-- 输入框 -->
10            <input type="text" name="title" required
11                lay-verify="required" placeholder="请输入标题"
12                autocomplete="off" class="layui-input"/>
13        </div>
14    </div>
15
16    <div class="layui-form-item">
17        <label class="layui-form-label">密码框区域</label>
18        <div class="layui-input-inline">
19            <input type="password" name="password" required
20                lay-verify="required" placeholder="请输入密码"
21                autocomplete="off" class="layui-input">
22        </div>
23    </div>
24 </form>
25
26 <script src="static/layui.js"></script>
27 <script type="text/javascript">
28     layui.use(['element', 'jquery', 'form'], function () {
29         var $ = layui.jquery
30         var form = layui.form
31     })
32 </script>
33

```

## 表单的常用属性

常用属性	描述
required	浏览器固定的必填字段
lay-verify	需要验证的类型(值为 required 的话表示必填项)
class="layui-input"	提供的通用的样式
class="layui-input-block"	占据全部宽度
class="layui-input-inline"	占据部分宽度
placeholder	当文本框为空时，默认显示的文本信息
autocomplete	表单元素是否自动填充(当浏览器中缓存中存在相同name属性的值时，会填充)

## 输入框

```

1 <!-- 输入框 -->
2 <input type="text" name="title" required lay-verify="required"
3   placeholder="请输入标题" autocomplete="off" class="layui-input" />
4

```

- required：注册浏览器所规定的必填字段
- lay-verify="required"：注册 form 模块需要验证的类型
- class="layui-input"：layui.css 提供的通用 CSS 类

## 下拉选择框

- 通过 selected 属性设置默认选中项
- 通过 disabled 属性开启禁用，可以设置 select 和 option 标签(表示禁用下拉框和禁用下拉选项)
- 通过 optgroup 标签给 select 分组
- 通过设置 lay-search 属性开启搜索匹配功能

```

1 <body>
2 <div class="layui-container">
3   <form action="" class="layui-form">
4     <div class="layui-form-item">
5       <label class="layui-form-label">城市</label>
6       <div class="layui-input-inline">
7         <!--
8           下拉选择框
9           1. 通过 selected 属性设置默认选中项
10          2. 通过 disabled 属性开启禁用，可以设置 select 和 option
11          标签（禁用下拉框和禁用下拉选项）
12          3. 可以通过 optgroup 标签给 select 分组
13          4. 通过设置 lay-search 属性开启搜索匹配功能
14        -->
15        <select name="city" lay-verify="required">
16          <option value="">请选择一个城市</option>
17          <option value="010">北京</option>
18          <option value="021" selected>上海</option>

```

```

18         <option value="0571" disabled>杭州</option>
19     </select>
20 </div>
21 </div>
22 </form>
23 </div>
24
25 <script src="static/layui.js"></script>
26 <script type="text/javascript">
27     layui.use(['jquery', 'form'], function () {
28         var $ = layui.jquery
29         var form = layui.form
30     })
31 </script>
32 </body>
33

```

## 分组

- 可以通过 optgroup 标签给select分组

```

1     <div class="layui-form-item">
2         <label class="layui-form-label">城市</label>
3         <div class="layui-input-inline">
4             <!-- 分组 -->
5             <select name="quiz">
6                 <option value="">请选择</option>
7                 <!-- 分组城市记忆 -->
8                 <optgroup label="城市记忆">
9                     <option value="你工作的第一个城市">你工作的第一个城市?
10                </optgroup>
11
12                <!-- 分组学生时代 -->
13                <optgroup label="学生时代">
14                    <option value="你的工号">你的工号? </option>
15                    <option value="你最喜欢的老师">你最喜欢的老师? </option>
16                </optgroup>
17            </select>
18        </div>
19    </div>
20

```

## 开启搜索匹配

- 通过设置 lay-search 属性开启搜索匹配功能

```

1      <div class="layui-form-item">
2          <label class="layui-form-label">城市</label>
3          <div class="layui-input-inline">
4              <!-- 开启搜索匹配 -->
5              <select name="city" lay-verify="" lay-search>
6                  <option value="">请选择</option>
7                  <option value="010">layer</option>
8                  <option value="021">form</option>
9                  <option value="0571">layim</option>
10             </select>
11         </div>
12     </div>
13

```

## 复选框

- 通过设置 title 属性设置自定义文本(如果不需要显示文本，则不设置 title 属性)
- 通过 checked 属性设置被选中的项
- 通过 lay-skin 属性设置复选框的样式效果(lay-skin="primary" 表示原始效果)
- 通过 disabled 属性禁用元素

```

1  <div class="layui-container">
2      <form action="" class="layui-form">
3          <div class="layui-form-item">
4              <label class="layui-form-label">爱好</label>
5              <div class="layui-input-block">
6                  <!--
7                      复选框
8                      1. 通过title属性设置自定义文本（如果不需要显示文本，则不设置
title属性）
9                      2. 通过checked属性设置被选中的项
10                     3. 通过lay-skin属性设置复选框的样式效果（lay-
skin="primary"表示原始效果）
11                     4. 通过disabled属性禁用元素
12                 -->
13                 <!-- 默认效果 -->
14                 <input type="checkbox" name="hobby" title="唱歌" checked
value="sing"/>
15                 <input type="checkbox" name="hobby" title="跳舞"
value="dance"/>
16                 <input type="checkbox" name="hobby" title="禁用" disabled/>
17                 <br>
18                 <!-- 原始效果 -->
19                 <input type="checkbox" name="hobby" title="唱歌" lay-
skin="primary" checked value="sing"/>
20                 <input type="checkbox" name="hobby" title="跳舞" lay-
skin="primary" value="dance"/>
21                 <input type="checkbox" name="hobby" title="禁用" lay-
skin="primary" disabled/>
22             </div>
23         </div>
24     </form>
25 </div>
26

```

## 开关

- 将复选框，设置 lay-skin="switch" 形成开关风格
- 通过 lay-text="打开的值|关闭的值" 来设定开关的两种状态的文本，通过| 分隔
- 通过 checked 属性设置默认打开状态
- 通过 disabled 属性禁用开关
- 通过 value 属性设置选中的值

```
1      <form action="" class="layui-form">
2          <div class="layui-form-item">
3              <label class="layui-form-label">开关</label>
4              <div class="layui-input-block">
5                  <!--
6                      开关
7                      将复选框，设置lay-skin="switch"形成开关风格
8                      1. 通过lay-text="打开的值|关闭的值"来设定开关的两种状态的文本，通过"|"分隔
9                      2. 通过checked设置默认打开状态
10                     3. 通过disabled属性禁用开关
11                     4. 通过value属性设置选中的值
12
13                 -->
14                 <input type="checkbox" name="aa" lay-skin="switch"/>
15                 <input type="checkbox" name="bb" lay-skin="switch" checked/>
16                 <input type="checkbox" name="cc" lay-skin="switch" checked
17 lay-text="on|off"/>
18                 <input type="checkbox" name="dd" lay-skin="switch" checked
19 lay-text="打开|关闭" value="打开"/>
20                 <input type="checkbox" name="ee" lay-skin="switch" lay-
21 text="打开|关闭" disabled/>
22             </div>
23         </div>
24     </form>
```

## 单选框

- 通过 checked 设置默认选中项
- 通过 disabled 属性禁用单选框
- 通过 value 属性设置选中的值

```
1      <div class="layui-container">
2          <form action="" class="layui-form">
3              <div class="layui-form-item">
4                  <label class="layui-form-label">性别</label>
5                  <div class="layui-input-block">
6                      <!--
7                          单选框
8                          1. 通过checked设置默认选中项
9                          2. 通过disabled属性禁用单选框
10                         3. 通过value属性设置选中的值
11
12                     -->
13                     <input type="radio" name="sex" value="nan" title="男">
14                     <input type="radio" name="sex" value="nv" title="女"
15 checked>
```

```

14         <input type="radio" name="sex" value="" title="中性"
disabled>
15     </div>
16 </div>
17 </form>
18 </div>
19

```

## 文本域

- 给 textarea 标签添加class="layui-textarea"

```

1 <div class="layui-container">
2     <form action="" class="layui-form">
3         <div class="layui-form-item">
4             <label class="layui-form-label">简介</label>
5             <div class="layui-input-inline">
6                 <!--
7                     文本域
8                     class="layui-textarea": layui.css提供的通用CSS类
9                 -->
10            <textarea name="" required lay-verify="required"
placeholder="请输入" class="layui-textarea"></textarea>
11        </div>
12    </div>
13 </form>
14 </div>

```

## 组装行内表单

行内表单：表单元素在一行显示（div 盒子也不会全部占完宽度）

- 给 div 设置 class="layui-inline"：定义外层行内
- 给 div 设置 class="layui-input-inline"：定义内层行内

```

1 <form action="" class="layui-form">
2     <div class="layui-form-item">
3         <!-- 定义外层行内 .layui-inline -->
4         <div class="layui-inline">
5             <label class="layui-form-label">范围</label>
6             <!-- 定义内层行内 .layui-input-inline -->
7             <div class="layui-input-inline" style="width: 100px;">
8                 <input type="text" name="price_min" placeholder="¥"
autocomplete="off" class="layui-input">
9             </div>
10            <div class="layui-form-mid">--</div>
11            <div class="layui-input-inline" style="width: 100px;">
12                <input type="text" name="price_max" placeholder="¥"
autocomplete="off" class="layui-input">
13            </div>
14        </div>
15        <!-- 定义外层行内 -->
16        <div class="layui-inline">
17            <label class="layui-form-label">密码</label>
18            <!-- 定义内层行内 -->
19            <div class="layui-input-inline" style="width: 100px;">

```

```

20         <input type="password" name="" autocomplete="off"
class="layui-input">
21     </div>
22 </div>
23 </div>
24 </form>

```

## 表单方框风格

- 给 form 标签追加 class="layui-form-pane ", 来设定表单的方框风格。
- 内部结构不变，值得注意的是：复选框/开关/单选框 这些组合在该风格下需要额外添加 pane属性

```

1  <!-- 通过追加 layui-form-pane 的class，来设定表单的方框风格。 -->
2  <form class="layui-form layui-form-pane" action="">
3      <!--
4      内部结构都一样，值得注意的是 复选框/开关/单选框 这些组合在该风格下需要额外添加
pane属性
5      -->
6      <div class="layui-form-item" pane>
7          <label class="layui-form-label">单选框</label>
8          <div class="layui-input-block">
9              <input type="radio" name="sex" value="男" title="男">
10             <input type="radio" name="sex" value="女" title="女" checked>
11          </div>
12      </div>
13 </form>

```