[Meta标签的属性设置： 2](#_Toc22979)

[一、基础样式的设置： 3](#_Toc1774)

[1、标题 3](#_Toc18904)

[2、副标题 4](#_Toc6918)

[3、内容区 4](#_Toc11820)

[4、底部 4](#_Toc9047)

[5、底部带有按钮 4](#_Toc12271)

[6、ionIC按钮 5](#_Toc7694)

[6.1图标按钮 5](#_Toc20333)

[6.2顶部、底部菜单按钮 5](#_Toc25039)

[6.3按钮栏 6](#_Toc10777)

[7、列表 7](#_Toc14876)

[7.1带头像的列表 9](#_Toc23011)

[7.2内嵌列表 10](#_Toc11212)

[7.3图片列表 10](#_Toc70)

[7.4输入框列表 11](#_Toc19052)

[7.5内联列表（带有输入框的提示文字） 13](#_Toc25260)

[7.6堆叠标签 14](#_Toc3460)

[7.7内嵌表单 14](#_Toc27426)

[7.8内嵌输入域 15](#_Toc19725)

[7.9带图标的输入框 16](#_Toc14186)

[8、切换开关 16](#_Toc1701)

[9、复选框 17](#_Toc29172)

[10、单选按钮 18](#_Toc22841)

[11、滑块 19](#_Toc19267)

[12、 下拉列表 19](#_Toc21318)

[13、 选项卡 19](#_Toc24517)

[13.1图标选项卡 20](#_Toc20450)

[13.2条纹选项卡 21](#_Toc29997)

[14、 网格 22](#_Toc16669)

[15、 颜色 27](#_Toc5062)

### Meta标签的属性设置：

**屏幕的缩放**  
content的几个属性：  
    width viewport的宽度[device-width | pixel\_value]width如果直接设置pixel\_value数值，大部分的**[安卓](http://lib.csdn.net/base/android" \o "Android知识库" \t "http://blog.csdn.net/kongjiea/article/details/_blank)**手机不支持，但是**[iOS](http://lib.csdn.net/base/ios" \o "iOS知识库" \t "http://blog.csdn.net/kongjiea/article/details/_blank)**支持；  
    height – viewport 的高度 （范围从 223 到 10,000 ）  
    user-scalable [yes | no]是否允许缩放  
    initial-scale [数值] 初始化比例（范围从 > 0 到 10）  
    minimum-scale [数值] 允许缩放的最小比例  
    maximum-scale [数值] 允许缩放的最大比例  
    target-densitydpi 值有以下（一般推荐设置中等响度密度或者低像素密度，后者设置具体的值dpi\_value，另外webkit内核已不准备再支持此属性）  
         -- dpi\_value 一般是70-400//没英寸像素点的个数  
         -- device-dpi设备默认像素密度  
         -- high-dpi 高像素密度  
         -- medium-dpi 中等像素密度  
         -- low-dpi 低像素密度

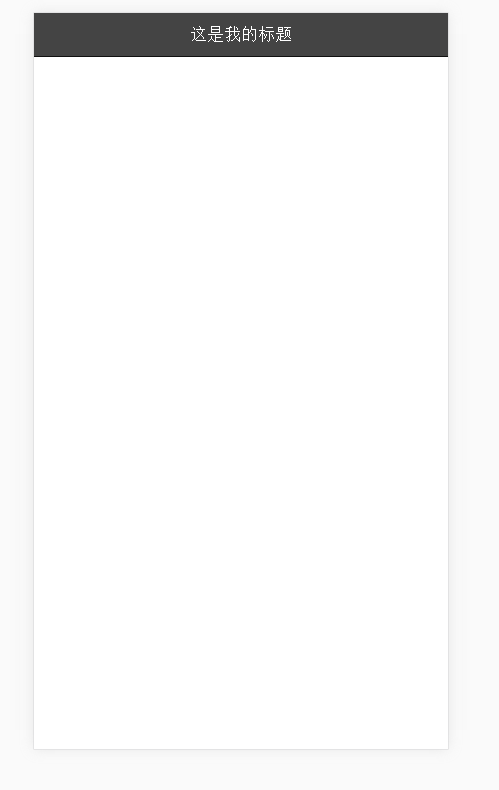
完整案例：

**<meta** name="viewport" content="width=device-width,height=device-height, user-scalable=no,initial-scale=1, minimum-scale=1, maximum-scale=1,target-densitydpi=device-dpi "**>**

### 一、基础样式的设置：

## 1、标题

<div class=”bar bar-header bar-light></div>（图1-1）



效果图（1-1）

标题有多种样式（主题）选择，选择不同的class样式可以同时改变背景色和里面的文字颜色，样式有以下几种：

bar-light

bar-stable

bar-positive

bar-calm

bar-balanced

bar-energized

bar-assertive

bar-royal

bar-dark

## 2、副标题

<div class=’bar bar-subheader bar-royal’>

<h2 class=’title’>副标题</h2>

</div>

## 3、内容区

<div class=’scroll-content has-header’></div>

（和低版本不同，现在需要使用scroll-content创建内容区，否则会出现中间内容显示不完整的bug，如果需要padding的话可以使用class=‘padding’添加内填充）

## 4、底部

底部和头部、副标题一样，只是把头部里面的header换成了footer

<div class=’bar bar-footer bar-calm’>

<h3 class=’title’>底部</h3>

</div>

## 5、底部带有按钮

<div class=’bar bar-footer bar-royal’>

<button class=’button button-clear’>左按钮</button>

<h3 class=’title’>标题</h3>

<button class="button button-clear">右按钮</button>

</div>

（当下面同时有标题和按钮时，左右按钮会自动左右分布，如果没有标题或者只想要右按钮时需要给按钮加上class=’pull-right’）

## 6、ionIC按钮

<button class="button">Default</button>

<button class="button button-light">light</button>

<button class="button button-stable">stable</button>

<button class="button button-positive">positive</button>

<button class="button button-calm">calm</button>

<button class="button button-balanced">balanced</button>

<button class="button button-energized">energized</button>

<button class="button button-assertive">assertive</button>

<button class="button button-royal">royal</button>

<button class="button button-dark">dark</button>

（必须class为button，后面的类名适用于不同的主题，如果使用button-clear可以清除按钮样式，只留文字）

（使用类名button-block可以使按钮的宽度设置为100%，即充斥整行）

（使用类名button-full可以使按钮充斥整行，比较类似于button-block，但button-full会移除margin）

（类名button-small和button-large可以使按钮变大或者缩小，同时会按比例缩小里面的内容）

（类名button-outline可以使按钮样式移除，比较类似于button-clear，但是button-outline会留下一个外边框）

### 6.1图标按钮

<button class="button icon-left ion-android-home">Home</button>

<button class="button icon-right ion-android-exit button-calm">Exit</button>

<button class="button icon ion-android-exit button-calm">Exit</button>

（使用class=’icon’创建一个图标按钮，通过ion-??-??可以选择在图标库中选择图标，当使用类名icon-right时按钮靠右，icon-left按钮靠左，如果不使用icon-right或icon-left的话按钮也可以正常显示，但是样式会比较小）

（使用button-calm可以调整按钮的主题，如果直接使用calm可以调整按钮文字和图标的颜色）

### 6.2顶部、底部菜单按钮

<div class="bar bar-footer">

<button class="button icon ion-android-apps"></button>

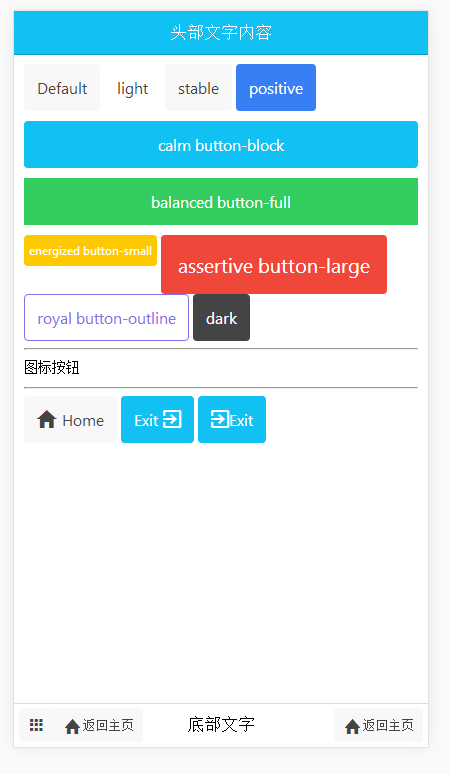
<h2 class="title">底部文字</h2>

<a href="" class="button icon ion-home">返回主页</a>

</div>

（在header、subheader或者footer可以创建按钮，按钮的标签可以为任何标签，a标签同样具有超级链接的效果，两个按钮时需要在文字的上下各放一个，按钮会自动左右对齐，如果只想有一个居右的按钮需要加上pull-right，三个以上按钮会出错）

（图1-2）



按钮样式（1-2）

### 6.3按钮栏

<div class="button-bar">

<button class="button button-calm">第一个</button>

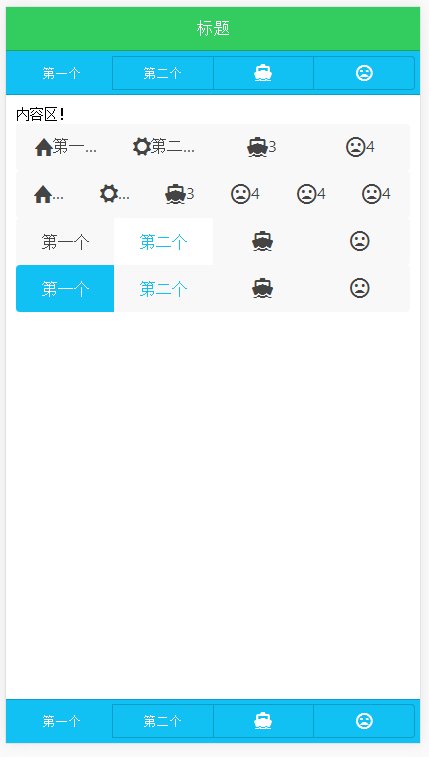
<button class="button calm">第二个</button>

<button class="button icon ion-android-boat"></button>

<button class="button icon ion-android-sad"></button>

</div>

（在任何地方都可以创建按钮栏，但如果是在页眉、页脚区创建则页眉页脚区文字会无法显示，按钮的宽度会根据数量自行缩放）（图1-3）



按钮栏（1-3）

## 7、列表

<ul class="list">

<li class="item">第1项</li>

<li class="item">第2项</li>

<li class="item">第3项</li>

<li class="item item-divider item-balanced">列表控件（带图标的列表）</li>

<li class="item item-icon-left"><i class="icon ion-alert-circled calm"></i>左边有图标</li>

<li class="item item-icon-right"><i class="icon ion-android-alert"></i>右边有图标</li>

<li class="item item-icon-left"><i class="icon ion-android-bookmark"></i>左边有图标右边有badge控件<span class="badge badge-calm">12</span></li>

<li class="item item-icon-left"><i class="icon ion-android-add-circle"></i>左边有图标右边有badge控件<span class="badge badge-royal">警告</span></li>

<li class="item item-icon-left item-icon-right balanced"><i class="icon balanced ion-android-add-circle"></i>两边都有图标<i class="icon ion-android-alarm-clock"></i></li>

<li class="item item-divider item-assertive">按钮列表</li>

<li class="item item-button-left">

<button class="button button-calm">

<i class="icon ion-android-checkbox-blank"></i>

</button>

左边带按钮的列表

</li>

<li class="item item-button-right">

<button class="button button-assertive">

<i class="icon ion-android-arrow-dropright-circle"></i>

</button>

右边带按钮的列表

</li>

<li class="item item-button-left item-button-right">

<button class="button button-assertive">

<i class="icon ion-android-arrow-dropright-circle"></i>

</button>

左右两边都带按钮的列表

<button class="button button-assertive">

<i class="icon ion-android-arrow-dropright-circle"></i>

</button>

</li>

</ul>

（使用list和item定义列表，没有标签限制）

（item为列表元素，如果想成为列表分隔符可以使用类名item-divider）

（带按钮的列表只能使用button标签）（图1-4）



列表（1-4）

### 7.1带头像的列表

<div class="scroll-content has-header">

<ul class="list">

<li class="item item-avatar">

<img src="img/1.png"/>

<h2>这里显示用户的名字</h2>

<p>这里是用户的简介</p>

</li>

<li class="item item-avatar">

<img src="img/2.png"/>

<h2>这里显示用户的名字</h2>

<p>这里是用户的简介</p>

</li>

<li class="item item-avatar">

<img src="img/3.png"/>

<h2>这里显示用户的名字</h2>

<p>这里是用户的简介</p>

</li>

<li class="item item-divider">略缩图列表</li>

<li class="item item-thumbnail-left">

<img src="img/12.png">

<h2>这是略缩图的名称</h2>

<p>这里是简介</p>

</li>

<li class="item item-thumbnail-right">

<img src="img/12.png">

<h2>这是略缩图的名称</h2>

<p>这里是简介</p>

</li>

</ul>

（在列表中添加item-avatar转换为带头像的列表，添加item-thumbnail转换为略缩图列表）

（这两种列表都只能放一张图片）

### 7.2内嵌列表

<ul class="list list-inset">

<li class="item item-divider">内嵌列表</li>

<li class="item">1111</li>

<li class="item">2222</li>

</ul>

（如果给content设置padding的话内嵌列表与普通列表无明显区别）

### 7.3图片列表

<ul class="list list-inset">

<li class="item item-divider">图片列表</li>

<li class="item item-image">

<img src="img/13.png"/>

</li>

</ul>

（使用item-image创建一个图片列表，里面只可以放一张图片，图片会自动按比例放大至100%宽度显示）

（使用ionic.bundle.js、声明angular模块ionic，并且将content的div标签名改为ion-content即可实现滚动效果）

### 7.4输入框列表

<ul class="list">

<li class="item item-input"><input type="text" placeholder="请输入账号："></li>

<li class="item item-input"><input type="password" placeholder="请输入密码"></li>

<li class="item item-input"><input type="email" placeholder="请输入邮箱"></li>

<li class="item item-input"><textarea class="no-resize">这是一个文本框</textarea></li>

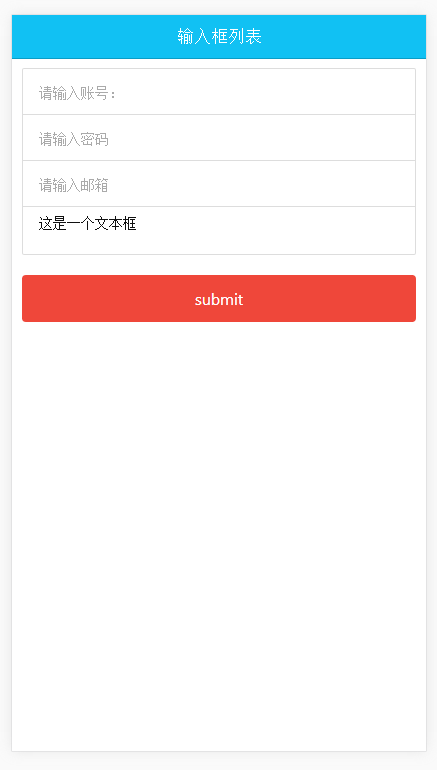
</ul>

<button class="button button-block button-assertive">submit</button>

（使用item-input创建一个输入框列表，里面只能有输入框不可以有提交按钮，提交按钮需要放在item之外，否则会出现位置偏右的问题）（图1-5/1-6）



Item中包含submit（图1-5）



使用button并放在item之外（图1-6）

### 7.5内联列表（带有输入框的提示文字）

<div class="scroll-content has-header">

<ul class="list">

<li class="item item-input padding">

<span class='input-label'>用户名：</span>

<input type="text">

</li>

<li class="item item-input padding">

<span class='input-label'>密码：</span>

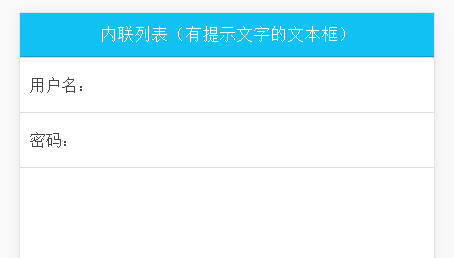
<input type="password">

</li>

</ul>

</div>

（在input标签前使用input-label可以为控件绑定提示文字，外层item的标签尽量使用label）（图1-7）



内联列表（图1-7）

### 7.6堆叠标签

<div class="list">

<label class="item item-input item-stacked-label">

<span class="input-label">用户名：</span>

<input type="text" placeholder="不能包含中文、特殊符号以及空格">

</label>

<label class="item item-input item-stacked-label">

<span class="input-label">密码：</span>

<input type="password" placeholder="请输入密码">

</label>

</div>

（使用item-stacked-label创建一个堆叠标签，和其他标签不一样的地方是堆叠标签的input-label在控件上方显示）

### 7.7内嵌表单

<div class="list list-inset">

<label class="item item-input">

<span class="input-label">用户名：</span>

<input type="text">

</label>

<label class="item item-input">

<span class="input-label">用户名：</span>

<input type="text">

</label>

<label class="item item-input">

<span class="input-label">用户名：</span>

<input type="text">

</label>

</div>

（使用list-inset定义内嵌表单，注意list-inset要放在list上）（图1-8）



内嵌表单（图1-8）

### 7.8内嵌输入域

<div class="list">

<div class="item item-input-inset">

<label class="item-input-wrapper">

<span class="input-label">内嵌输入域：</span>

<input type="text">

</label>

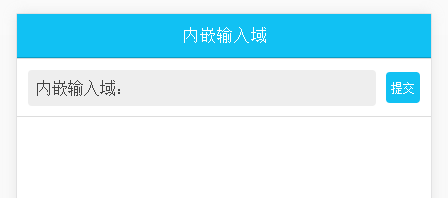
<button class="button button-calm button-small">提交</button>

</div>

</div>

（使用item-input-inset定义一个内嵌输入域）

（注意：item-input-inset需要放在item上，item-input-wrapper需要放在label里面）



内嵌输入域（图1-9）

### 7.9带图标的输入框

<div class="list list-inset">

<label class="item item-input">

<i class="icon ion-android-add-circle placeholder-icon"></i>

<input type="text" placeholder="这是提示文字">

</label>

</div>

（在label前加一个icon，附类名placeholder-icon可以在input的placeholder前面加入一个图标）



带图标的输入框（图1-10）

## 8、切换开关

<label class="toggle">

<input type="checkbox">

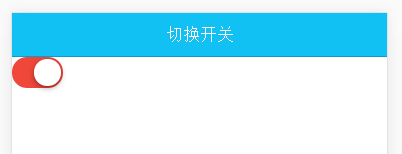
<div class="track">

<div class="handle"></div>

</div>

</label>

（使用以上结构来制定一个ionIC仿ios的切换开关）



切换开关（图1-11）

8.1切换开关列表

<li class="item item-toggle">

这是我的文字内容

<label class="toggle toggle-assertive">

<input type="checkbox" checked="checked">

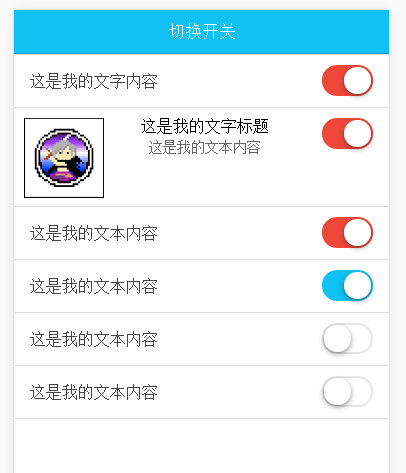
<div class="track">

<div class="handle"></div>

</div>

</label>

</li>



切换开关列表（图1-12）

## 9、复选框

<li class="item item-checkbox item-checkbox-right">

这是复选框的练习

<label class="checkbox">

<input type="checkbox">

</label>

</li>

（使用以上结构来创建一个带有复选框的列表，checkbox默认在列表左侧，如果想让CheckBox出现在右侧需要在item上面加上class=’item-checkbox-right’）



复选框（图1-13）

## 10、单选按钮

<label class="item item-radio">

<input type="radio" name="a">

<div class="radio-content">

<div class="item-content">

这是我的内容

</div>

<i class="radio-icon ion-checkmark"></i>

</div>

</label>

（通过上述内容实现一组单选按钮，单选按钮需要用同样的name才可以实现单选，注意item-content外面还需要套一层radio-content，否则不会出现效果，在菜鸟教程中此处是错误的）



单选按钮（图1-14）

## 11、滑块

<li class="item range range-assertive">

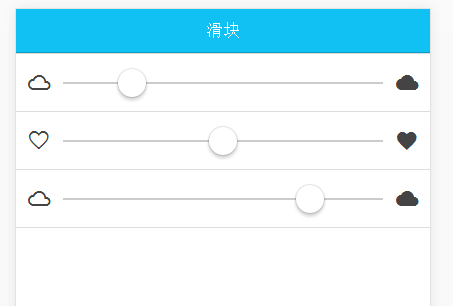
<i class="icon ion-android-favorite-outline"></i>

<input type="range">

<i class="icon ion-android-favorite"></i>

</li>

（通过上述代码实现滑块，注意两个i标签必须分别在控件的上下，否则会挤到一起）



滑块（图1-15）

## 下拉列表

<li class="item item-input item-select">

<p class="input-label">

这是下拉列表左边的文字内容

</p>

<select>

<option>第一个</option>

<option>第二个</option>

<option>第三个</option>

<option>第四个</option>

</select>

</li>

（QQ浏览器模拟手机时下拉菜单会出现问题）

## 选项卡

<div class="tabs">

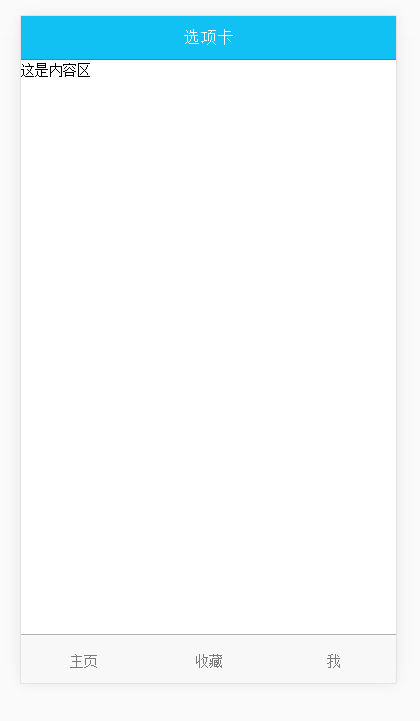
<a class='tab-item' href="#">主页</a>

<a class='tab-item' href="#">收藏</a>

<a class='tab-item' href="#">我</a>

</div>

（选项卡会默认在页面的最下方，如果有选项卡的话就不可以有bar-footer）（图1-16）



Tabs选项卡（图1-16）

### 13.1图标选项卡

<div class="tabs">

<a class='tab-item' href="#">

<i class='icon ion-alert-circled'></i>

</a>

<a class='tab-item' href="#">

<i class='icon ion-home'></i>

</a>

<a class='tab-item' href="#">

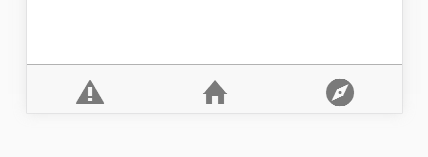
<i class='icon ion-android-compass'

</a>

</div>

（图1-17）

（通过给tabs加类名tabs-icon-top或tabs-icon-left可以实现不同位置的图标与文字混排显示）（图1-18）



图标选项卡（图1-17）





图文混排（图1-18）

### 13.2条纹选项卡

<div class="tabs-striped tabs-color-assertive">

<div class="tabs">

<a href="" class="tab-item active">

<i class='icon ion-android-add'></i>

</a>

<a href="" class="tab-item">

<i class='icon ion-android-bicycle'></i>

</a>

<a href="" class="tab-item">

<i class='icon ion-android-locate'></i>

</a>

</div>

</div>

（使用tabs-striped创建一个条纹样式的选项卡，注意要在这个标签内部创建一个tabs来在选项按钮的容器；tabs-striped可以通过tabs-backgroundcolor-[color]和tabs-color-[color]来设置背景色和文字颜色，通过给tab-item附加class=’active’使选项卡处于活跃状态的样式）

## 网格

<div class="row">

<div class="col">col</div>

<div class="col">col</div>

<div class="col">col</div>

</div>

（使用col拼接的网格会自动适应并平分剩余的空间）（图1-19）

<div class="row">

<div class="col col-10">col-10</div>

<div class="col col-10">col-10</div>

<div class="col col-80">col-80</div>

</div>

（使用col-\*的网格会按照比例分配空间，多余的部分留白，可以与col网格一起使用）（图1-20）

<div class="row">

<div class="col col-10">col-10</div>

<div class="col col-40 col-offset-50">offset-50</div>

</div>

（使用col-offset-\*的网格会按照比例位移）（图1-21）

|  |  |
| --- | --- |
| .col-10 | 10% |
| .col-20 | 20% |
| .col-25 | 25% |
| .col-33 | 33.3333% |
| .col-50 | 50% |
| .col-67 | 66.6666% |
| .col-75 | 75% |
| .col-80 | 80% |
| .col-90 | 90% |

Col-\*具体百分比（图1-19）

|  |  |
| --- | --- |
| .col-offset-10 | 10% |
| .col-offset-20 | 20% |
| .col-offset-25 | 25% |
| .col-offset-33 | 33.3333% |
| .col-offset-50 | 50% |
| .col-offset-67 | 66.6666% |
| .col-offset-75 | 75% |
| .col-offset-80 | 80% |
| .col-offset-90 | 90% |

偏移的具体百分比（图1-20）



网格（图1-21）

**14.1**垂直对齐网格

<div class="row">

<div class="col">col</div>

<div class="col">col</div>

<div class="col">col</div>

<div class="col">1<br />2<br />3<br />4</div>

</div>

<div class="row">

<div class="col col-top">col-top</div>

<div class="col col-center">col-center</div>

<div class="col col-bottom">col-bottom</div>

<div class="col">1<br />2<br />3<br />4</div>

</div>

（通过使用col-top、col-center、col-bottom实现网格的不同基线的对正）（图1-22）



垂直对齐网格（图1-22）

**14.2**响应式网格

<div class="row responsive-sm">

<div class="col">col</div>

<div class="col">col</div>

<div class="col">col</div>

<div class="col">col</div>

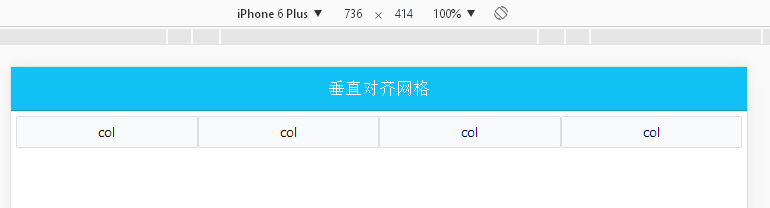
</div>

（创建一个响应式网格，在不同分辨率下会显示出不同的样式）（图1-23）

| 响应式类 | 描述 |
| --- | --- |
| .responsive-sm | 小于手机横屏 |
| .responsive-md | 小于平板竖屏 |
| .responsive-lg | 小于平板横屏 |



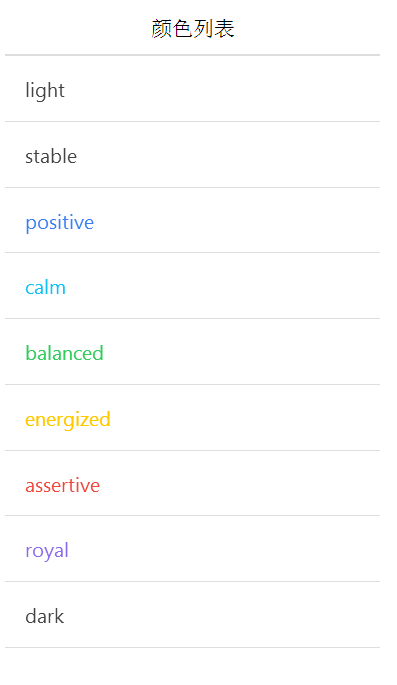
手机竖屏



手机横屏

（图1-23）

## 颜色



颜色列表