# Bootstrap组件

**目 录**

[Bootstrap组件 1](#_Toc23064)

[1下拉菜单 4](#_Toc22128)

[1.1下拉菜单（基本用法） 4](#_Toc14400)4

1.dropdown父容器包含整个下拉菜单

2.给触发元素加上dropdown-toggle类以及自定义元素data-toggle="dropdown"

3.ul/ol>li 加上dropdown-menu

[1.2下拉菜单（原理分析） 5](#_Toc3508)

[1.3 下拉菜单（下拉分隔线） 7](#_Toc24154)

divlder

[1.4 下拉菜单（菜单标题） 7](#_Toc29778)

dropdown-header

[1.5 下拉菜单（对齐方式） 8](#_Toc27100)

dropdwon-menu-right

pull-right

[1.6下拉菜单（菜单项状态） 9](#_Toc13658)

active 选中

disibel 禁用

[2 按钮 10](#_Toc5953)

[2.1按钮（按钮组） 11](#_Toc2649)

把多个按钮放在一个组里面，就是按钮组.

第一个和最后一个圆角，而且按钮之间没有间距的粘合在一起

btn-group

[2.2按钮（按钮工具栏） 11](#_Toc29912)

btn-toolbar

一个按钮工具栏里面，放置多个按钮组。

[运行效果如下所示： 13](#_Toc32630)

[2.3按钮（嵌套分组） 14](#_Toc249)

就是把容器的class从dropdown变成btn-group

[2.4 按钮（垂直分组） 15](#_Toc22860)

btn-group-vertical

[2.5 按钮的向下向上三角形 16](#_Toc14059)

caret

如果要下三角变成上三角，那么就要把dropdown容器，改为dropup

[3 导航 17](#_Toc12983)

[3.1 导航（基础样式） 17](#_Toc26630)

nav

[3.2 导航（标签形tab导航） 18](#_Toc23026)

nav nav-tabs

[3.3导航（胶囊形(pills)导航） 19](#_Toc27347)

nav nav-pills

[3.4 导航（垂直堆叠的导航） 20](#_Toc28596)

nav nav-pills nav-stacked

[3.5 自适应导航（使用） 21](#_Toc14389)

nav nav-pills nav-justified

[3.6 导航加下拉菜单（二级导航） 22](#_Toc8242)

1.把li当做dropdown父容器

2.把a标签当做触发元素,.dropdwon-toggle和属性data-toggle="dropdown"

3.在里面嵌套dropdown-menu

[3.7 面包屑式导航 24](#_Toc3817)

ul.Breadcrumb>li>a

[4输入框组 24](#_Toc19758)

只是用来给表单元素添加额外样式

input-group

.input-group>span.input-group-addon{$}+input[type="text"]

[4.1基本实例 24](#_Toc14291)

[4.2不同尺寸的按钮组 25](#_Toc31097)

input-group input-group-lg

[4.3作为额外元素的多选框和单选框 26](#_Toc6648)

.input-group>(span.input-group-addon>input[type="radio"])+input[type="text"]

[4.4作为额外元素的按钮 27](#_Toc2453)

.input-group>(span.input-group-btn>button[type="button"])+input[type="text"]

[5导航栏 27](#_Toc796)

[5.1 导航条基础 27](#_Toc19036)

[5.2 基础导航条 28](#_Toc29263)

nav.navbar navbar-default/inverse

[5.3为导航条添加标题、二级菜单及状态 28](#_Toc30158)

navbar-header父容器

navbar-brand 表示标题或者logo

ul.nav.navbar-nav>li>a 菜单

active disibel 状态

[5.4 带表单的导航条 30](#_Toc23542)

navbar-form 注意：表单元素不会自动漂浮

配合navbar-form-left/right

[5.5 导航条中的按钮、文本和链接 32](#_Toc28543)

navbar-btn 按钮

navbar-text 文本

navbar-link 链接

[5.5 固定导航条 35](#_Toc23317)

navbar-fixed-top 固定在上方

navbar-fixed-bottom 固定在下方

[5.6 响应式导航条 36](#_Toc32052)

第一步：把要折叠的部分放在collapse navbar-collapse 容器里面，并且为这个容器加上一个id。

第二步：在navbar-header里面定义一个button，并且加上.navbar-toggle类，加上自定义属性data-toggle="collapse" 和 data-target="#折叠区的id"。

button.navbar-toggle[type="button"][data-toggle="collapse"][data-target="#折叠区的id"]>.icon-bar\*3。

[6 分页 39](#_Toc8672)

[6.1 分页导航（带页码的分页导航） 39](#_Toc9653)

ul.pagination>li>a

[6.2分页导航（翻页分页导航） 41](#_Toc19421)

ul.pager>li>a

li.prve.....左漂浮

li.next右漂浮

[7 标签 42](#_Toc28381)

label label-info

[8 徽章 44](#_Toc1573)

bedge 一般放在一段文字的末尾，或用定位来定位到想要的位置。

# 1下拉菜单

## 1.1下拉菜单（基本用法）

在Bootstrap框架中的下拉菜单组件是一个独立的组件

在使用Bootstrap框架的下拉菜单时，必须调用Bootstrap框架提供的bootstrap.js文件。

**特别声明：**因为Bootstrap的组件交互效果都是依赖于jQuery库写的插件，所以在使用bootstrap.min.js之前一定要先加载jquery.min.js才会生效果。

<div class="dropdown">

<button class="btn btn-default dropdown-toggle" type="button" id="dropdownMenu1" data-toggle="dropdown">

下拉菜单

<span class="caret"></span>

</button>

<ul class="dropdown-menu" role="menu" aria-labelledby="dropdownMenu1">

<li role="presentation"><a role="menuitem" tabindex="-1" href="#">下拉菜单项</a></li>

…

<li role="presentation" class="divider"></li>

<li role="presentation"><a role="menuitem" tabindex="-1" href="#">下拉菜单项</a></li>

</ul>

</div>

****使用方法：****

在使用Bootstrap框架中的下拉菜单组件时，其结构运用的正确与否非常的重要，如果结构和类名未使用正确，直接影响组件是否能正常运用。

**1、**使用一个名为“dropdown”的容器包裹了整个下拉菜单元素，示例中为:

<div class="dropdown"></div>

**2、**使用了一个<button>按钮做为父菜单，并且定义类名“dropdown-toggle”和自定义“data-toggle”属性，且值必须和最外容器类名一致，此示例为:

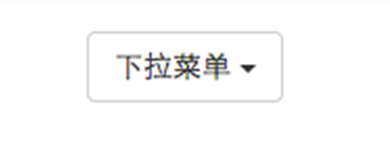
data-toggle="dropdown"

**3、**下拉菜单项使用一个ul列表，并且定义一个类名为“dropdown-menu”，此示例为:

<ul class="dropdown-menu">

## 1.2下拉菜单（原理分析）

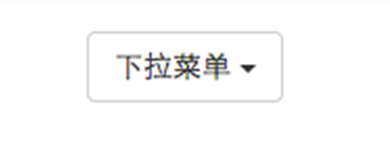
Bootstrap框架中的下拉菜单组件，其下拉菜单项默认是隐藏的，如下所示：

[](http://img.mukewang.com/53e1f1850001230803900164.jpg)

因为“dropdown-menu”默认样式设置了“display:none”

[](http://img.mukewang.com/53e1f3dc0001c10204150412.jpg)

当用户再次点击时，下拉菜单将继续隐藏，如下所示：

[](http://img.mukewang.com/53e1f1850001230803900164.jpg)

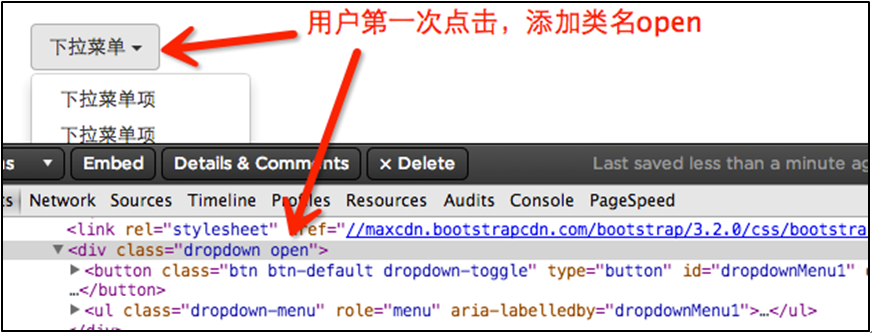
****原理分析：****

现在我们来分析一下实现原理，非常简单，通过js技术手段，给父容器“div.dropdown”添加或移除类名“open”来控制下拉菜单显示或隐藏。也就是说，默认情况，“div.dropdown”没有类名“open”，当用户第一次点击时，“div.dropdown”会添加类名“open”；当用户再次点击时，“div.dropdown”容器中的类名“open”又会被移除。我们可以通过浏览器的firebug查看整个过程：

**默认情况：**

[](http://img.mukewang.com/53e314020001537208700352.jpg)

**用户第一次点击：**

[](http://img.mukewang.com/53e314360001a81808720333.jpg)

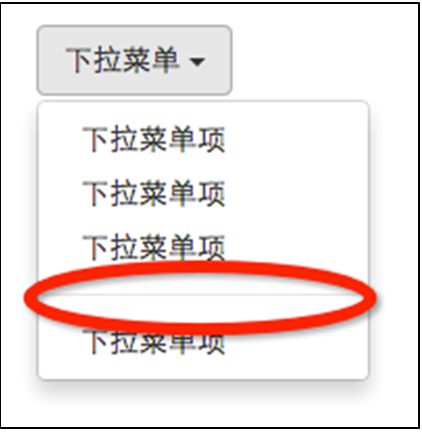
**用户再次点击：**

[](http://img.mukewang.com/53e31461000180e608710333.jpg)

## 1.3 下拉菜单（下拉分隔线）

在Bootstrap框架中的下拉菜单还提供了**下拉分隔线**，假设下拉菜单有两个组，那么组与组之间可以通过添加一个空的<li>，并且给这个<li>添加类名“divider”来实现添加下拉分隔线的功能。对应的样式代码：

效果如下：

[](http://img.mukewang.com/53e346260001aed304220432.jpg)

## 1.4 下拉菜单（菜单标题）

为了让分组更明显，还可以给每个组添加一个头部（标题）。如下：

<div class="dropdown">

<button class="btn btn-default dropdown-toggle" type="button" id="dropdownMenu1" data-toggle="dropdown">

下拉菜单

<span class="caret"></span>

</button>

<ul class="dropdown-menu" role="menu" aria-labelledby="dropdownMenu1">

<li role="presentation" class="dropdown-header">第一部分菜单头部</li>

<li role="presentation"><a role="menuitem" tabindex="-1" href="#">下拉菜单项</a></li>

…

<li role="presentation" class="divider"></li>

<li role="presentation" class="dropdown-header">第二部分菜单头部</li>

…

<li role="presentation"><a role="menuitem" tabindex="-1" href="#">下拉菜单项</a></li>

</ul>

</div>

运行效果如下：

[](http://img.mukewang.com/53e34b1e0001ccdd07440651.jpg)

## 1.5 下拉菜单（对齐方式）

****实现右对齐方法：****

Bootstrap框架中下拉菜单默认是左对齐，如果你想让下拉菜单相对于父容器右对齐时，可以在“dropdown-menu”上添加一个“pull-right”或者“dropdown-menu-right”类名，如下所示：

<div class="dropdown">

<button class="btn btn-default dropdown-toggle" type="button" id="dropdownMenu1" data-toggle="dropdown">

下拉菜单

<span class="caret"></span>

</button>

<ul class="dropdown-menu pull-right" role="menu" aria-labelledby="dropdownMenu1">

…

</ul>

</div>

上面代码中的“pull-right”类可以用“dropdown-menu-right”代替，两个类的作用是一样的。

运行效果如下所示：

[](http://img.mukewang.com/53e34c370001522204970469.jpg)

**下拉菜单与父容器左边对齐:**

与此同时，还有一个类名刚好与“dropdown-menu-right”相反的类名“dropdown-menu-left”，其效果就是让下拉菜单与父容器左边对齐，其实就是默认效果。

## 1.6下拉菜单（菜单项状态）

下拉菜单项的默认的状态**（不用设置）有悬浮状态（:hover）**和**焦点状态（:focus）**：

下拉菜单项除了上面两种状态，还有**当前状态（.active）**和**禁用状态（.disabled）**。这两种状态使用方法只需要在对应的菜单项上添加对应的类名：

<div class="dropdown">

<button class="btn btn-default dropdown-toggle" type="button" id="dropdownMenu1" data-toggle="dropdown">

下拉菜单

<span class="caret"></span>

</button>

<ul class="dropdown-menu" role="menu" aria-labelledby="dropdownMenu1">

<li role="presentation" class="active"><a role="menuitem" tabindex="-1" href="#">下拉菜单项</a></li>

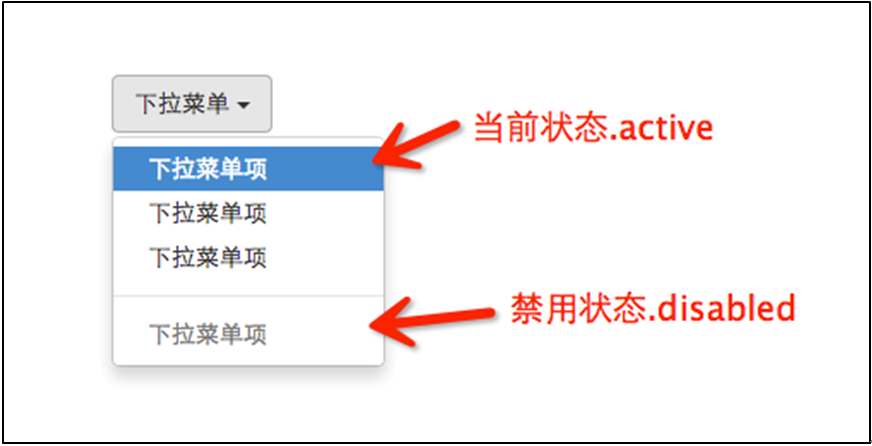
….

<li role="presentation" class="disabled"><a role="menuitem" tabindex="-1" href="#">下拉菜单项</a></li>

</ul>

</div>

运行效果如下：

[](http://img.mukewang.com/53e44d0d000131d208720446.jpg)

# 2 按钮

## 2.1按钮（按钮组）

单个按钮在Web页面中的运用有时候并不能满足我们的业务需求，常常会看到将多个按钮组合在一起使用，比如富文本编辑器里的一组小图标按钮等。

**使用方法：**

按钮组和下拉菜单组件一样，需要依赖于button.js插件才能正常运行。不过我们同样可以直接只调用bootstrap.js文件。因为这个文件已集成了button.js插件功能。对于结构方面，非常的简单。使用一个名为“btn-group”的容器，把多个按钮放到这个容器中。如下所示：

<div class="btn-group">

<button type="button" class="btn btn-default">

<span class="glyphicon glyphicon-step-backward"></span>

</button>

…

<button type="button" class="btn btn-default">

<span class="glyphicon glyphicon-step-forward"></span>

</button>

</div>

运行效果如下所示：

[](http://img.mukewang.com/53e458d10001af8808310131.jpg)

除了可以使用<button>元素之外，还可以使用其他标签元素，比如<a>标签。唯一要保证的是：不管使用什么标签，“.btn-group”容器里的标签元素需要带有类名“.btn”。

## 2.2按钮（按钮工具栏）

在富文本编辑器中，将按钮组分组排列在一起，比如说复制、剪切和粘贴一组；左对齐、中间对齐、右对齐和两端对齐一组，如下图所示：

[](http://img.mukewang.com/53e45edc00019ad308600101.jpg)

那么Bootstrap框架按钮工具栏也提供了这样的制作方法，你只需要将按钮组“btn-group”按组放在一个大的容器“btn-toolbar”中，如下所示：

<div class="btn-toolbar">

<div class="btn-group">

…

</div>

<div class="btn-group">

…

</div>

<div class="btn-group">

…

</div>

<div class="btn-group">

…

</div>

</div>

实现原理主要是让容器的多个分组“btn-group”元素进行浮动，并且组与组之前保持5px的左外距。

**运行效果如下：**

[](http://img.mukewang.com/53e462020001bd2e08240084.jpg)

**按钮组大小设置**

我们知道按钮是通过btn-lg、btn-sm和btn-xs三个类名来调整padding、font-size、line-height和border-radius属性值来改变按钮大小。那么按钮组的大小，我们也可以通过类似的方法：

**☑  .btn-group-lg:大按钮组**

**☑  .btn-group-sm:小按钮组**

**☑  .btn-group-xs:超小按钮组**

只需要在“.**btn-group”类名上追加对应的类名，就可以得到不同大小的按钮组**。如下所示：

<div class="btn-toolbar">

<div class="btn-group btn-group-lg">

…

</div>

<div class="btn-group">

…

</div>

<div class="btn-group btn-group-sm">

…

</div>

<div class="btn-group btn-group-xs">

…

</div>

</div>

## 运行效果如下所示：

[](http://img.mukewang.com/53e4632b0001bb2808230100.jpg)

## 2.3按钮（嵌套分组）

很多时候，我们常把下拉菜单和普通的按钮组排列在一起，实**现类似于导航菜单的效果**。如下所示：

[](http://img.mukewang.com/53e466ac0001273008410307.jpg)

使用的时候，只需要**把当初制作下拉菜单的“dropdown”的容器换成“btn-group”，并且和**普通的按钮放在同一级。如下所示：

<div class="btn-group">

<button class="btnbtn-default" type="button">首页</button>

<button class="btnbtn-default" type="button">产品展示</button>

<button class="btnbtn-default" type="button">案例分析</button>

<button class="btnbtn-default" type="button">联系我们</button>

<div class="btn-group">

<button class="btnbtn-default dropdown-toggle" data-toggle="dropdown" type="button">关于我们<span class="caret"></span></button>

<ul class="dropdown-menu">

<li><a href="##">公司简介</a></li>

<li><a href="##">企业文化</a></li>

<li><a href="##">组织结构</a></li>

<li><a href="##">客服服务</a></li>

</ul>

</div>

</div>

## 2.4 按钮（垂直分组）

前面看到的示例，按钮组都是水平显示的。但在实际运用当中，总会碰到垂直显示的效果。在Bootstrap框架中也提供了这样的风格。我们**只需要把水平分组的“btn-group”类名换成“btn-group-vertical”即可**。如下所示：

<div class="btn-group-vertical">

<button class="btnbtn-default" type="button">首页</button>

<button class="btnbtn-default" type="button">产品展示</button>

<button class="btnbtn-default" type="button">案例分析</button>

<button class="btnbtn-default" type="button">联系我们</button>

<div class="btn-group">

<button class="btnbtn-default dropdown-toggle" data-toggle="dropdown" type="button">关于我们<span class="caret"></span></button>

<ul class="dropdown-menu">

<li><a href="##">公司简介</a></li>

<li><a href="##">企业文化</a></li>

<li><a href="##">组织结构</a></li>

<li><a href="##">客服服务</a></li>

</ul>

</div>

</div>

运行的效果如下：

[](http://img.mukewang.com/53e85b8f0001cfd001870309.jpg)

## 2.5 按钮的向下向上三角形

按钮的向下三角形，我们是通过在<button>标签中添加一个“<span>”标签元素，并且命名为“caret”:

<button class="btn btn-default dropdown-toggle" data-toggle="dropdown" type="button">按钮下拉菜单<span class="caret"></span></button>

这个三角形完全是通过CSS代码来实现的,另外在按钮中的三角形“caret”做了一定的样式处理：

.btn .caret {

margin-left: 0;

}

.btn-lg .caret {

border-width: 5px 5px 0;

border-bottom-width: 0;

}

.dropup .btn-lg .caret {

border-width: 0 5px 5px;

}

有的时候我们的下拉菜单会向上弹起，这个时候我们的三角方向需要朝上显示，实现方法：需要在“.btn-group”类上追加“dropup”类名（这也是做向上弹起下拉菜单要用的类名）。

<div class="btn-group dropup">

<button class="btn btn-default dropdown-toggle" data-toggle="dropdown" type="button">按钮下拉菜单<span class="caret"></span></button>

<ul class="dropdown-menu">

<li><a href="##">按钮下拉菜单项</a></li>

<li><a href="##">按钮下拉菜单项</a></li>

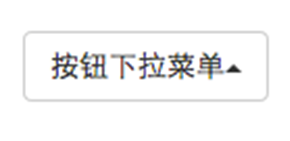
<li><a href="##">按钮下拉菜单项</a></li>

<li><a href="##">按钮下拉菜单项</a></li>

</ul>

</div>

**运行效果如下：**

[](http://img.mukewang.com/53e8651e0001c0a102900141.jpg)

# 3 导航

## 3.1 导航（基础样式）

导航对于一位前端人员来说并不陌生。可以说导航是一个网站重要的元素组件之一，可以便于用户查找网站所提供的各项功能服务。导航的制作方法也是千奇百怪，五花八门。在这一节中将向大家介绍如何使用Bootstrap框架制作各式各样的导航。

**导航基础样式**

Bootstrap框架中制作导航条主要通过“.nav”样式。默认的“.nav”样式不提供默认的导航样式，必须附加另外一个样式才会有效，比如“nav-tabs”、“nav-pills”之类。

## 3.2 导航（标签形tab导航）

标签形导航，也称为选项卡导航。特别是在很多内容分块显示的时，使用这种选项卡来分组十分适合。

标签形导航是通过“nav-tabs”样式来实现。**在制作标签形导航时需要在原导航“nav”上追加此类名**，如：

<ul class="nav nav-tabs">

<li><a href="##">Home</a></li>

<li><a href="##">CSS3</a></li>

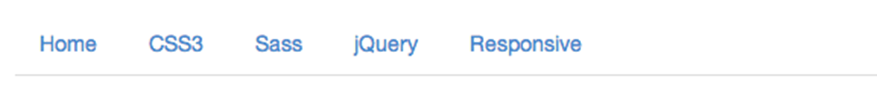
<li><a href="##">Sass</a></li>

<li><a href="##">jQuery</a></li>

<li><a href="##">Responsive</a></li>

</ul>

运行的效果如下所示：

[](http://img.mukewang.com/53e86aa60001805308940091.jpg)

实现原理非常的简单，将菜单项（li）按块显示，并且让他们在同一水平上排列，然后定义非高亮菜单的样式和鼠标悬浮效果。

其实上例的效果和我们平时看到的选项卡效果并不一致。一般情况之下，选项卡教会有一个当前选中项。其实在Bootstrap框架也相应提供了。**假设我们想让“Home”项为当前选中项，只需要在其标签上添加类名“class="active"”**即可：

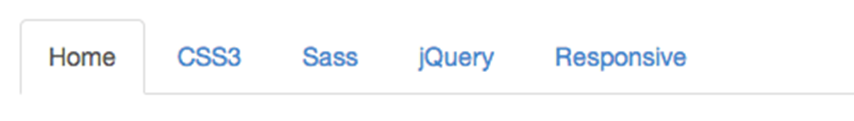
<ul class="nav nav-tabs">

<li class="active"><a href="##">Home</a></li>

…

</ul>

运行效果如下：

[](http://img.mukewang.com/53e86b7700019e0308540121.jpg)

除了当前项之外，有的选项卡还带有禁用状态，实现这样的效果，只需要在标签项上添加“class="disabled"”即可：

<ul class="nav nav-tabs">

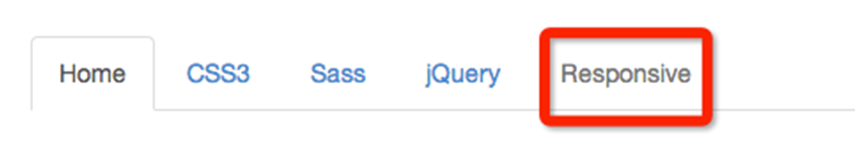
<li class="active"><a href="##">Home</a></li>

…

<li class="disabled"><a href="##">Responsive</a></li>

</ul>

运行效果如下：

[](http://img.mukewang.com/53e86c8b00015ca208550155.jpg)

## 3.3导航（胶囊形(pills)导航）

胶囊形（pills）导航听起来有点别扭，因为其外形看起来有点像胶囊形状。但其更像我们平时看到的大众形导航。当前项高亮显示，并带有圆角效果。其实现方法和“nav-tabs”类似，同样的结构，**只需要把类名“nav-tabs”换成“nav-pills”**即可：

<ul class="nav nav-pills">

<li class="active"><a href="##">Home</a></li>

<li><a href="##">CSS3</a></li>

<li><a href="##">Sass</a></li>

<li><a href="##">jQuery</a></li>

<li class="disabled"><a href="##">Responsive</a></li>

</ul>

[](http://img.mukewang.com/53e86ee60001711e08160307.jpg)

## 3.4 导航（垂直堆叠的导航）

在实际运用当中，除了水平导航之外，还有垂直导航，就类似前面介绍的垂直排列按钮一样。**制作垂直堆叠导航只需要在“nav-pills”的基础上添加一个“nav-stacked”类名**即可：

<ul class="nav nav-pills nav-stacked">

<li class="active"><a href="##">Home</a></li>

<li><a href="##">CSS3</a></li>

<li><a href="##">Sass</a></li>

<li><a href="##">jQuery</a></li>

<li class="disabled"><a href="##">Responsive</a></li>

</ul>

运行效果如下：

[](http://img.mukewang.com/53e871a2000102b707240444.jpg)

垂直堆叠导航与胶囊形导航相比，主要是让导航项不浮动，让其垂直排列，然后给相邻导航项留有一定的间距。

在下拉菜单一节中，下拉菜单组与组之间有一个分隔线。其实在垂直堆叠导航也具有这样的效果，**只需要添加在导航项之间添加“<li class=”nav-divider”></li>”**即可：

<ul class="nav nav-pills nav-stacked">

<li class="active"><a href="##">Home</a></li>

<li><a href="##">CSS3</a></li>

<li><a href="##">Sass</a></li>

<li><a href="##">jQuery</a></li> <li class="nav-divider"></li>

<li class="disabled"><a href="##">Responsive</a></li>

</ul>

## 3.5 自适应导航（使用）

自适应导航指的是导航占据容器全部宽度，而且菜单项可以像表格的单元格一样自适应宽度。**在制作自适应导航时更换使用类名“nav-justified”。当然他需要和“nav-tabs”或者“nav-pills”配合在一起使用**。如：

<ul class="nav nav-tabs nav-justified">

<li class="active"><a href="##">Home</a></li>

<li><a href="##">CSS3</a></li>

<li><a href="##">Sass</a></li>

<li><a href="##">jQuery</a></li>

<li><a href="##">Responsive</a></li>

</ul>

运行效果如下：

[IMG_256](http://img.mukewang.com/53ed99aa00016bcb08630061.jpg)

## 3.6 导航加下拉菜单（二级导航）

前面介绍的都是使用Bootstrap框架制作一级导航，但很多时候，在Web页面中是离不开二级导航的效果。那么在Bootstrap框架中制作二级导航就更容易了。**只需要将li当作父容器，使用类名“dropdown”，同时在li中嵌套另一个列表ul**，使用前面介绍下拉菜单的方法就可以：

<ul class="nav nav-pills">

<li class="active"><a href="##">首页</a></li>

<li class="dropdown">

<a href="##" class="dropdown-toggle" data-toggle="dropdown">教程<span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a href="##">CSS3</a></li>

…

</ul>

</li>

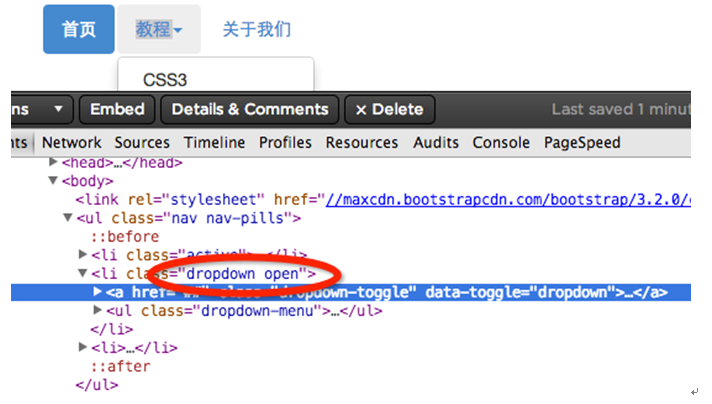
<li><a href="##">关于我们</a></li>

</ul>

运行效果如下：

[](http://img.mukewang.com/53e877e700014b0104150304.jpg)

通过浏览器调试工具，不难发现，点击有二级导航的菜单项，会自动添加“open”类名，再次点击就会删除添加的“open”类名：

[](http://img.mukewang.com/53e878580001e0b307120404.jpg)

简单点来说，就是依靠这个类名来控制二级导航显示与否.

在制作下拉菜单时，可以用分隔线，那么在二级导航中是否可以呢？**只需要添加“<li class=”nav-divider”></li>”**这样的一个空标签就可以了。

运行效果如下：

[](http://img.mukewang.com/53e878b600013f7d04750337.jpg)

## 3.7 面包屑式导航

面包屑(Breadcrumb)一般用于导航，主要是起的作用是告诉用户现在所处页面的位置（当前位置）。在Bootstrap框架中面包屑也是一个独立模块组件：

使用方法：

使用方式就很简单，**为ol加入breadcrumb类**：

<ol class="breadcrumb">

<li><a href="#">首页</a></li>

<li><a href="#">我的书</a></li>

<li class="active">《图解CSS3》</li>

</ol>

## 4输入框组

## 4.1基本实例

通过在文本输入框 <input> 前面、后面或是两边加上文字或按钮，可以实现对表单控件的扩展。**为 .input-group 赋予 .input-group-addon 类，可以给 .form-control 的前面或后面添加额外的元素。**

<div class="input-group">

<span class="input-group-addon" >@</span>

<input type="text" class="form-control" placeholder="Username" >

</div>

<div class="input-group">

<input type="text" class="form-control" placeholder="Recipient'>

<span class="input-group-addon">@example.com</span>

</div>

<div class="input-group">

<span class="input-group-addon">$</span>

<input type="text" class="form-control" >

<span class="input-group-addon">.00</span>

</div>



## 4.2不同尺寸的按钮组

为 .input-group 添加相应的尺寸类，其内部包含的元素将自动调整自身的尺寸。不需要为输入框组中的每个元素重复地添加控制尺寸的类，只需要在input-group后面追加input-group-lg/md/sm则可控制整个组输入框的大小。

<div class="input-group input-group-lg">

<span class="input-group-addon" >@</span>

<input type="text" class="form-control" placeholder="Username" >

</div>

<div class="input-group input-group-md">

<input type="text" class="form-control" placeholder="Recipient">

<span class="input-group-addon">@example.com</span>

</div>

<div class="input-group input-group-sm">

<span class="input-group-addon">$</span>

<input type="text" class="form-control" >

<span class="input-group-addon">.00</span>

</div>



## 4.3作为额外元素的多选框和单选框

如果要使用单选按钮或者是复选按钮作为额外元素，实现方式是十分的简单，直接在带带有input-group-addon类的元素里面写上表单元素即可。

<div class="row">

<div class="col-lg-6">

<div class="input-group">

<span class="input-group-addon">

<input type="checkbox" >

</span>

<input type="text" class="form-control" >

</div><!-- /input-group -->

</div><!-- /.col-lg-6 -->

<div class="col-lg-6">

<div class="input-group">

<span class="input-group-addon">

<input type="radio" >

</span>

<input type="text" class="form-control" >

</div><!-- /input-group -->

</div><!-- /.col-lg-6 -->

</div><!-- /.row -->



## 4.4作为额外元素的按钮

为输入框组添加按钮需要额外添加一层嵌套，**不是 .input-group-addon，而是添加 .input-group-btn 来包裹按钮元素**。由于不同浏览器的默认样式无法被统一的重新赋值，所以才需要这样做。

<div class="input-group input-group-lg">

<span class="input-group-btn" >

<button class="btn btn-default" type="button">详细信息</button>

</span>

<input type="text" class="form-control" placeholder="Username" >

</div>



# 5导航栏

## 5.1 导航条基础

导航条（navbar）和导航（nav），就相差一个字，多了一个“条”字。其实在Bootstrap框架中他们还是明显的区别。在导航条(navbar)中有一个背景色、而且导航条可以是纯链接（类似导航），也可以是表单，还有就是表单和导航一起结合等多种形式。在这一节中将一起探讨Bootstrap框架中导航条的使用。

## 5.2 基础导航条

在Bootstrap框中，导航条和导航从外观上差别不是太多，但在实际使用中导航条要比导航复杂得多。我们先来看导航条中最基础的一个——基础导航条。

**使用方法：**

在制作一个基础导航条时，主要分以下几步：

**第一步：添加一个容器（div），并且使用类名“navbar”和“navbar-default”**

**第二步：首先在制作导航的列表(<ul class=”nav”>)基础上添加类名“navbar-nav”**

“.navbar”样式的主要功能就是设置左右padding和圆角等效果，但他和颜色相关的样式没有进行任何的设置。

## 5.3为导航条添加标题、二级菜单及状态

**加入导航条标题**

在Web页面制作中，常常在菜单前面都会有一个标题（文字字号比其它文字稍大一些），其实在Bootstrap框架也为大家做了这方面考虑，**其通过“navbar-header”和“navbar-brand”来实现**。

<div class="navbar navbar-default">

<div class="navbar-header">

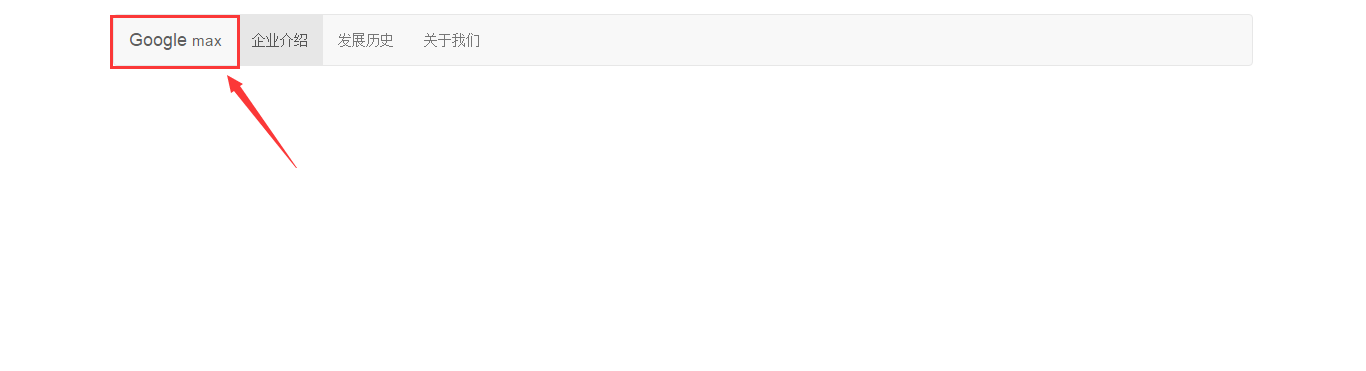
<a href="#" class="navbar-brand">Google <small>max</small></a>

</div>

</div>

**原理分析：**

此功能主要起一个提醒功能，当然改良一下可以当作是logo(此处不做过多阐述)。其样式主要是加大了字体设置，并且设置了最大宽度：



**导航条状态、二级菜单**

同样的，在基础导航条中对菜单提供了当前状态，禁用状态，悬浮状态等效果，而且也可以带有二级菜单的导航条,效果图如下：



代码如下：

<div class="navbar navbar-default">

<div class="navbar-header">

<a href="#" class="navbar-brand">Google <small>max</small></a>

</div>

<ul class="nav navbar-nav">

<li class="active"><a href="">企业介绍</a></li>

<li class="dropdown">

<a href="#" class="dropdown-toggle" data-toggle="dropdown">

发展历史

<span class="caret"></span>

</a>

<ul class="dropdown-menu">

<li><a href="">1990年</a></li>

<li><a href="">1996年</a></li>

<li><a href="">2000年</a></li>

<li><a href="">2008年</a></li>

<li><a href="">2014年</a></li>

</ul>

</li>

<li class="disabled"><a href="">关于我们</a></li>

</ul>

</div>

## 5.4 带表单的导航条

有的导航条中会带有搜索表单，比如新浪微博的导航条：

[IMG_256](http://img.mukewang.com/53edcf9d00013cf506950056.jpg)

在Bootstrap框架中提供了一个**“navbar-form”，使用方法很简单，在navbar容器中放置一个带有navbar-form类名的表单。**

<nav class="navbar navbar-default" role="navigation">

<div class="container">

<div class="navbar-header">

<a href="##" class="navbar-brand">软融</a>

</div>

<ul class="nav navbar-nav">

<li class="active">

<a href="##">网站首页</a>

</li>

<li class="dropdown">

<a href="##" data-toggle="dropdown" class="dropdown-toggle">系列教程<span class="caret"></span></a>

<ul class="dropdown-menu">

<li>

<a href="##">CSS3</a>

</li>

<li>

<a href="##">JavaScript</a>

</li>

<li class="disabled">

<a href="##">PHP</a>

</li>

</ul>

</li>

<li>

<a href="##">名师介绍</a>

</li>

<li>

<a href="##">成功案例</a>

</li>

<li>

<a href="##">关于我们</a>

</li>

</ul>

<form action="##" class="navbar-form">

<div class="form-group">

<input type="text" class="form-control" placeholder="请输入关键词" />

</div>

<button type="submit" class="btn btn-default">搜索</button>

</form>

</div>

</nav>



## 5.5 导航条中的按钮、文本和链接

Bootstrap框架的导航条中除了使用navbar-brand中的a元素和navbar-nav的ul和navbar-form之外，还可以使用其他元素。框架提供了三种其他样式：

**1、****导航条中的按钮navbar-btn**

**2、导航条中的文本navbar-text**

**3、导航条中的普通链接navbar-link**

**在设置这三个元素之前，可以使用漂浮pull-left把表单漂浮起来**

**但这三种样式在框架中使用时受到一定的限制，需要和navbar-brand、navbar-nav配合起来使用。而且对数量也有一定的限制，一般情况在使用一到两个不会有问题，超过两个就会有问题。**

**导航条中的文本navbar-text及导航条中的普通链接navbar-link**

div class="navbar navbar-default">

<div class="navbar-header">

<a href="##" class="navbar-brand">Bootstrap</a>

</div>

<div class="nav navbar-nav">

<a href="##" class="navbar-text">Navbar Text</a>

<a href="##" class="navbar-text">Navbar Text</a>

<a href="##" class="navbar-text">Navbar Text</a>

</div>

</div>

<div class="navbar navbar-default" role="navigation">

<div class="navbar-header">

<a href="##" class="navbar-brand">Bootstrap</a>

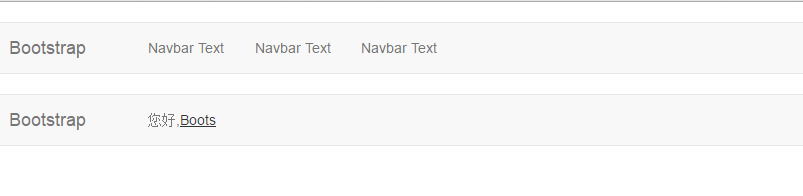
</div>

<div class="nav navbar-nav">

<p class="navbar-text">您好,

<a href="#" class="navbar-link">Boots</a>

</p>

</div> </div>

**导航条中的按钮navbar-btn**

<nav class="navbar navbar-default" role="navigation">

<div class="container">

<div class="navbar-header">

<a href="##" class="navbar-brand">软融教育</a>

</div>

<ul class="nav navbar-nav">

<li class="active">

<a href="##">网站首页</a>

</li>

<li class="dropdown">

<a href="##" data-toggle="dropdown" class="dropdown-toggle">系列教程<span class="caret"></span></a>

<ul class="dropdown-menu">

<li>

<a href="##">CSS3</a>

</li>

<li>

<a href="##">JavaScript</a>

</li>

<li class="disabled">

<a href="##">PHP</a>

</li>

</ul>

</li>

<li>

<a href="##">名师介绍</a>

</li>

<li>

<a href="##">成功案例</a>

</li>

<li>

<a href="##">关于我们</a>

</li>

</ul>

<a href="#" class="btn btn-primary btn-sm navbar-btn navbar-right hidden-xs">订阅课程</a>

<form action="##" class="navbar-form navbar-ri">

<div class="form-group">

<input type="text" class="form-control" placeholder="请输入关键词" />

</div>

<button type="submit" class="btn btn-default">搜索</button>

</form>

</div>

</nav>



## 5.5 固定导航条

很多情况之一，设计师希望导航条固定在浏览器顶部或底部，这种固定式导航条的应用在移动端开发中更为常见。Bootstrap框架提供了两种固定导航条的方式：

**☑  .navbar-fixed-top：导航条固定在浏览器窗口顶部**

**☑  .navbar-fixed-bottom：导航条固定在浏览器窗口底部**

使用方法很简单，只需要在制作导航条最外部容器navbar上追加对应的类名即可：

<div class="navbar navbar-default navbar-fixed-top" role="navigation">

　…

</div>

<div class="content">我是内容</div>

<div class="navbar navbar-default navbar-fixed-bottom" role="navigation">

　…

</div>

**实现原理：**

实现原理很简单，就是在navbar-fixed-top和navbar-fixed-bottom使用了position：fixed属性，并且设置navbar-fixed-top的top值为0,而navbar-fixed-bottom的bottom值为0。

**存在bug及解决方法:**

从运行效果中大家不难发现，页面主内容顶部和底部都被固定导航条给遮住了。为了避免固定导航条遮盖内容，我们需要在body上做一些处理：

body {

padding-top: 70px;/\*有顶部固定导航条时设置\*/

padding-bottom: 70px;/\*有底部固定导航条时设置\*/

}

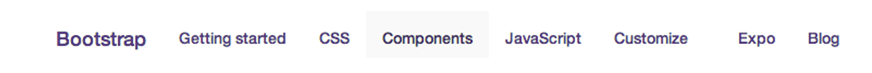
因为固定导航条默认高度是50px，**我们一般设置padding-top和padding-bottom的值为70px**，当然有的时候还是需要具体情况具体分析。

## 5.6 响应式导航条

如今浏览Web页面的终端不在是一尘不变了，前面示例实现的导航条仅能适配于大屏幕的浏览器，但当浏览器屏幕变小的时候，就不适合了。因此响应式设计也就随之而来。那么在一个响应式的Web页面中，对于响应式的导航条也就非常的重要。例如Bootstrap框架官网的导航条：

[IMG_256](http://img.mukewang.com/53eded3b0001db2a06970046.jpg)

（宽屏时效果）

[](http://img.mukewang.com/53f580af00017ef408720073.jpg)

（中屏时效果）

[](http://img.mukewang.com/53f580e30001bba208690079.jpg)

（窄屏时效果）

**使用方法：**

1、保证在**窄屏时需要折叠的内容必须包裹在带一个div内，并且为这个div加入collapse、navbar-collapse两个类名。最后为这个div添加一个class类名或者id名**。

需要折叠的div代码段：

<div class="collapse navbar-collapse" id="example">

<ul class="nav navbar-nav">

…

</ul>

</div>

**2、保证在窄屏时要显示的图标样式（固定写法）：**

<div class="navbar-header">

<button class="navbar-toggle" data-toggle="collapse" data-target="#example">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a href="#" class="navbar-brand"><strong>google</strong>.net</a>

</div>

**3、并为button添加data-target=".类名/#id名"，究竞是类名还是id名呢？由需要折叠的div来决定。如：**

<button class="navbar-toggle" type="button" data-toggle="collapse" data-target="#example">

...

</button>

也可以这么写，需要折叠的div代码段：

<div class="collapse navbar-collapse example" >

<ul class="nav navbar-nav">

…

</ul>

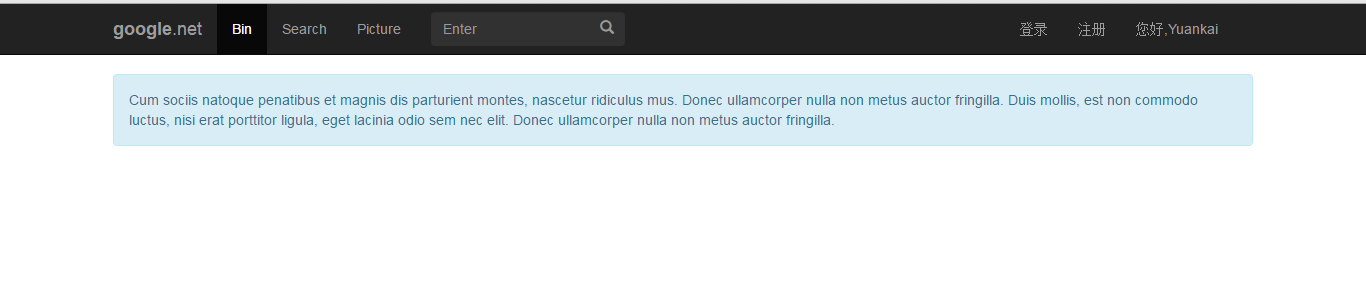
</div>

窄屏时要显示的图标：

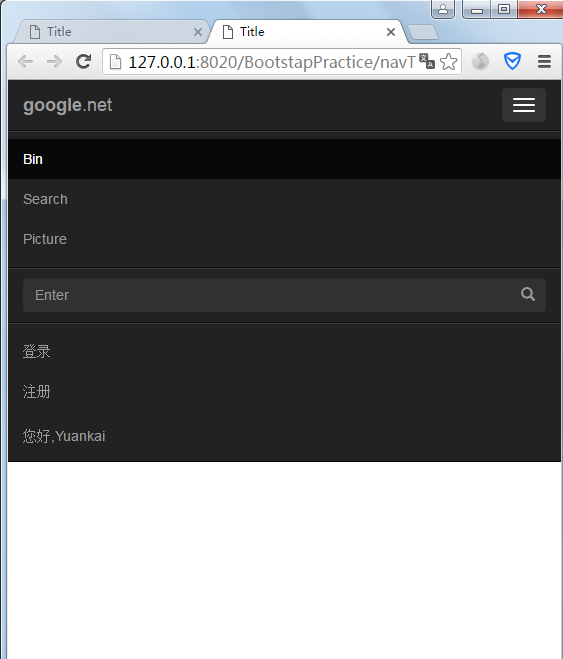
<button class="navbar-toggle" type="button" data-toggle="collapse" data-target=".example">

...

</button>

**宽屏时效果：**

**窄屏时效果:**



# 6 分页

## 6.1 分页导航（带页码的分页导航）

分页导航几乎在哪个网站都可见。好的分页导航能给用户带来更好的用户体验。在Bootstrap框架中提供了两种分页导航：

**☑   带页码的分页导航 1234**

**☑   带翻页的分页导航 prev next**

**带页码的分页导航**

带页码的分页导航，可能是最常见的一种分页导航，特别是在列表页内容超多的时候，会给用户提供分页的导航方式。

**使用方法：**

平时很多同学喜欢用div>a和div>span结构来制作带页码的分页导航。不过，在Bootstrap框架中使用的是ul>li>a这样的结构，在ul标签上加入**pagination**方法：

<ul class="pagination">

<li><a href="#">&laquo;</a></li>

<li><a href="#">1</a></li>

<li><a href="#">2</a></li>

<li><a href="#">3</a></li>

<li><a href="#">4</a></li>

<li><a href="#">5</a></li>

<li><a href="#">&raquo;</a></li>

</ul>

运行效果：

[](http://img.mukewang.com/53f5972900018aa605480162.jpg)

**实现原理：**

从效果中可以看出，当前状态页码会高亮显示，而且不能点击。而最后一页是禁用状态，也不能点击。

**大小设置：**

在Bootstrap框架中，也可以通过几个不同的情况来设置其大小。类似于按钮一样：

**1、通过“pagination-lg”让分页导航变大；**

**2、通过“pagination-sm”让分页导航变小：**

<ul class="pagination pagination-lg">

…

</ul>

<ul class="pagination">

…

</ul>

<ul class="pagination pagination-sm">

…

</ul>



## 6.2分页导航（翻页分页导航）

Bootstrap框架除了提供带页码的分页导航之外还提供了翻页导航。这种分页导航常常在一些简单的网站上看到，比如说个人博客，杂志网站等。这种分页导航是看不到具体的页码，只会提供一个“上一页”和“下一页”的按钮。

**使用方法：**

在实际使用中，翻页分页导航和带页码的分页导航类似，为ul标签加入**pager**类：

<ul class="pager">

<li><a href="#">&laquo;上一页</a></li>

<li><a href="#">下一页&raquo;</a></li>

</ul>

**对齐样式设置：**

默认情况之下，翻页分页导航是居中显示，但有的时候我们需要一个居左，一个居右。Bootstrap框架提供了两个样式：

**☑   previous：让“上一步”按钮居左**

**☑   next：让“下一步”按钮居右**

具体使用的时候，只需要在li标签上添加对应类名即可：

<ul class="pager">

<li class="****previous****"><a href="#">&laquo;上一页</a></li>

<li class="****next****"><a href="#">下一页&raquo;</a></li>

</ul>

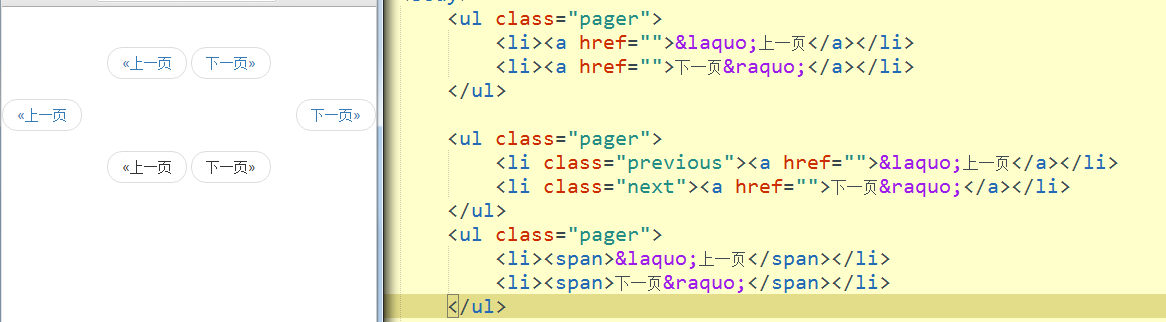
**状态样式设置：**

和带页码分页导航一样，**如果在li标签上添加了disabled类名的时候，分页按钮处于禁用状态，但同样不能禁止其点击功能。你可以通过js来处理，或将a标签换成span标签**。

<ul class="pager"> ****<li class="disabled"><span>&laquo;上一页</span></li>****

<li><a href="#">下一页&raquo;</a></li>

</ul>



# 7 标签

在一些Web页面中常常会添加一个标签用来告诉用户一些额外的信息，比如说在导航上添加了一个新导航项，可能就会加一个“new”标签，来告诉用户。这是新添加的导航项。如下图所示：

[](http://img.mukewang.com/53f5a3810001256d05550068.jpg)

那么在Bootstrap框架中特意将这样的效果提取出来成为一个标签组件，并且以“.label”样式来实现高亮显示。

**使用原理：**

使用方法很简单，你可以在使用span这样的行内标签：

<h3>Example heading ****<span class="label label-default">New</span>****</h3>

**颜色样式设置：**

和按钮元素button类似，label样式也提供了多种颜色：

**☑   label-deafult:默认标签，深灰色**

**☑   label-primary：主要标签，深蓝色**

**☑   label-success：成功标签，绿色**

**☑   label-info：信息标签，浅蓝色**

**☑   label-warning：警告标签，橙色**

**☑   label-danger：错误标签，红色**

主要是通过这几个类名来修改背景颜色和文本颜色：

<span class="label label-default">默认标签</span>

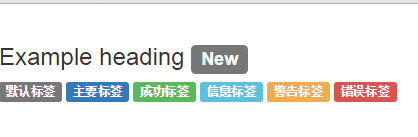
<span class="label label-primary">主要标签</span>

<span class="label label-success">成功标签</span>

<span class="label label-info">信息标签</span>

<span class="label label-warning">警告标签</span>

<span class="label label-danger">错误标签</span>



# 8 徽章

从某种意义上来说，徽章效果和前面介绍的标签效果是极其的相似。也是用来做一些提示信息使用。常出现的是一些系统发出的信息，比如你登录你的twitter后，如果你信息没有看，系统会告诉你有多少信息未读，如下图所示：

[](http://img.mukewang.com/53f5aac500010a7f04370079.jpg)

在Bootstrap框架中，把这种效果称作为徽章效果，使用**“badge”**样式来实现。

**使用方法：**

使用方法，其实也没什么太多可说的，你可以像标签一样，使用span标签来制作，然后为他加入badge类：

<a href="#">Inbox <span class="badge">42</span></a>

正如开头所说，可以将徽章与按钮或者导航之类配合使用：

<div class="navbar navbar-default" role="navigation">

　<div class="navbar-header">

　 <a href="##" class="navbar-brand">软融教育</a>

　</div>

<ul class="nav navbar-nav">

<li class="active"><a href="##">网站首页</a></li>

<li><a href="##">系列教程</a></li>

<li><a href="##">名师介绍</a></li>

<li><a href="##">成功案例****<span class="badge">23</span>****</a></li>

<li><a href="##">关于我们</a></li>

</ul>

</div>

**按钮和胶囊形导航设置徽章：**

另外，徽章在按钮元素button和胶囊形导航nav-pills也可以有类似的样式，只不过是颜色不同而以：

<ul class="nav nav-pills">

<li class="active"><a href="#">Home ****<span class="badge">42</span>****</a></li>

…

<li><a href="#">Messages****<span class="badge">3</span>****</a></li>

</ul>

<ul class="navnav-pills nav-stacked" style="max-width: 260px;">

<li class="active">

<a href="#"> ****<span class="badge pull-right">42</span>****

Home

</a>

</li>

…

<li>

<a href="#"> ****<span class="badge pull-right">3</span>****

Messages

</a>

</li>

</ul>

<button class="btnbtn-primary" type="button">

Messages ****<span class="badge">4</span>****

</button>