# Bootstrap组件-2

**目 录**

[Bootstrap组件-2 1](#_Toc4579)

[1 缩略图 3](#_Toc31327)

[1.1 缩略图（一） 3](#_Toc15597)

**thumbnail 结合row栅格系统来使用，一般用于商品排列**

[1.2 缩略图（二） 5](#_Toc30446)

.thumbnail>.caption

[2警示框 6](#_Toc8923)

[2.1 警示框--默认警示框 6](#_Toc13431)

.alert.alert-info

[2.2 警示框--可关闭的警示框 8](#_Toc5610)

**第一步:**给父容器加上.alert.alert-info alert-dimissabel

**第二步:**写上一个button给button.colse[data-dismiss="alert"]{&times;}

[2.3 警示框--警示框的链接 9](#_Toc27499)

<p>零落<a class="alert-link">成泥</a><p>

[3 进度条 11](#_Toc1866)

[3.1 进度条--基本样式 12](#_Toc181)

**第一步**：创建一个div.progress

**第二步**：div.progress>div.progress-bar[style="width:40%;"]

[3.2 进度条--彩色进度条 13](#_Toc1742)

给div.progress-bar追加.progress-bar-info/success/danger/wanig

[3.2 进度条--动态条纹进度条 14](#_Toc20774)

**第一步**：创建一个div.progress

**第二步**：div.progress>div.progress-bar[style="width:40%;"]

**第三步：**给div.progress追加.porgress-stirped和active

[3.3 进度条--层叠进度条 15](#_Toc289)

一个progress下面可以有多个progress-bar，可以组合在一起。

**注意：progress-bar不要超过100%，要不然最后一个div看不见。**

[3.4 进度条--带Label的进度条 17](#_Toc5465)

[4 媒体对象 18](#_Toc14504)

[4.1 媒体对象--默认媒体对象 19](#_Toc2437)

**媒体对象的容器：media**

**媒体对象的对象：media-object**

**媒体对象的主体：media-body**

**媒体对象的标题：media-heading**

在使用的时候，常常会把media-object漂浮起来（pull-left,pull-right）。

[4.2 媒体对象--媒体对象列表 21](#_Toc5756)

**ul.media-list>li.media\*10**

5看文字

4看视频

3看图片

2自己动手

1教别人

[5 列表组 22](#_Toc8476)

[5.1 列表组--基础列表组 22](#_Toc7197)

ul.list-group>li.list-group-item>a

[5.2 列表组--带徽章的列表组 23](#_Toc3113)

bedge在列表组里面会发生向右漂浮

[5.3 列表组--带链接的列表组 25](#_Toc17355)

[5.4 列表组--自定义列表组 27](#_Toc17605)

.list-group-item>.list-group-item-heading+.list-group-item-text

[5.5 列表组--列表项的状态设置 27](#_Toc19629)

active disibeld

[5.6 列表组--多彩列表组 28](#_Toc19745)

.list-group-item.list-group-item-info

[6 面板 30](#_Toc6447)

[6.1 面板--基础面板 30](#_Toc19191)

父容器panel panel-success/info/danger

[6.2 面板--带有头和尾的面板 30](#_Toc14648)

panel-heading 头部

panel-body 身体

panel-footer 尾部

[6.3 面板--彩色面板 31](#_Toc21341)

只会改变头部的颜色模式

[6.4 面板--面板中嵌套列表组 32](#_Toc18215)

panel-body>.div.list-group>li.list-group-item

[7 模态弹出框（Modals） 35](#_Toc4685)

# 1 缩略图

## 1.1 缩略图（一）

缩略图在网站中最常用的地方就是产品列表页面，一行显示几张图片，有的在图片底下（左侧或右侧）带有标题、描述等信息。Bootstrap框架将这一部独立成一个模块组件。**并通过“thumbnail”样式配合bootstrap的网格系统来实现**。可以将产品列表页变得更好看。

**使用方法：**

**通过“thumbnail”样式配合bootstrap的网格系统来实现。**

前面也说过了，缩略图的实现是配合网格系统一起使用，假设我们一个产品列表，如下图所示：

[](http://img.mukewang.com/5418e97a00014d6806620159.jpg)

先来看结构：

<div class="container">

<div class="row">

<div class="col-sm-3 col-xs-6 thumbnail">

<img src="http://placehold.it/170x180" alt="170x180"/>

</div>

<div class="col-sm-3 col-xs-6 thumbnail">

<img src="http://placehold.it/170x180" alt="170x180"/>

</div>

<div class="col-sm-3 col-xs-6 thumbnail">

<img src="http://placehold.it/170x180" alt="170x180"/>

</div>

<div class="col-sm-3 col-xs-6 thumbnail">

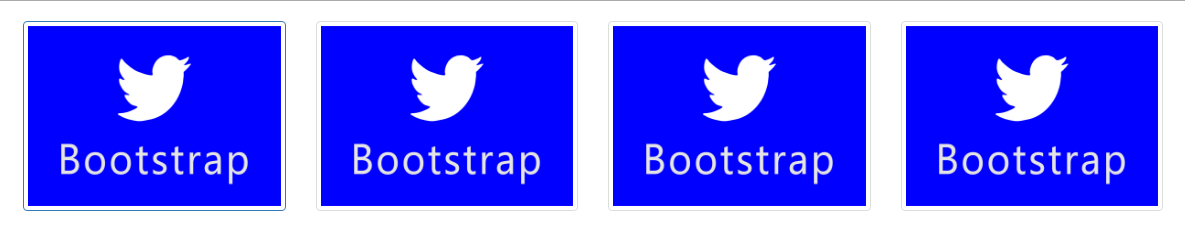
<img src="http://placehold.it/170x180" alt="170x180"/>

</div>

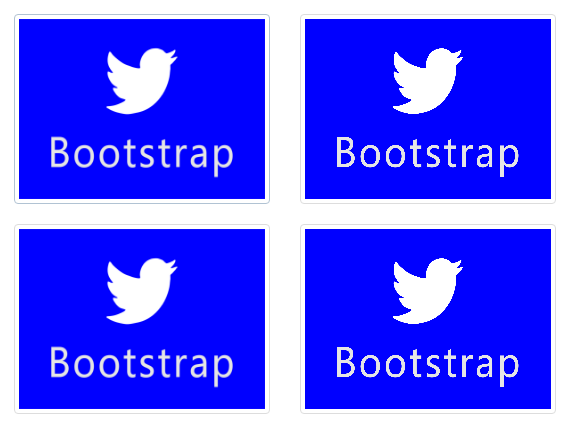
</div>

</div>

上面的结构表示的是在宽屏幕（可视区域大于768px）的时候，一行显示四个缩略图



在窄屏（可视区域小于768px）的时候，一行只显示两个缩略图：



## 1.2 缩略图（二）

上一小节，展示的仅只有缩略图的一种使用方式，除了这种方式之外，还可以让缩略图配合标题、描述内容，按钮等：



在仅有缩略图的基础上，**添加了一个div名为“caption“的容器，在这个容器中放置其他内容，比如说标题，文本描述，按钮**等：

<div class="col-sm-3 col-xs-6">

<div class="thumbnail">

<img src="logo.jpg" style="width: 100%;height: 180px;"/>

<div class="caption">

<h3>Thumbnail label</h3>

<p>Cras justo odio, dapibus ac facilisis in, egestas eget quam. Donec id elit non mi porta gravida at eget metus. Nullam id dolor id nibh ultricies vehicula ut id elit.</p>

<button class="btn btn-default">

button

</button>

<button class="btn btn-primary">

button

</button>

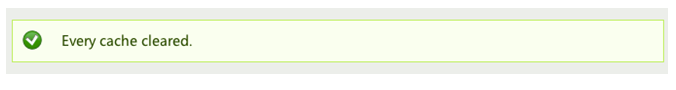
</div>

</div>

</div>

# 2警示框

在网站中，网页总是需要和用户一起做沟通与交流。特别是当用户操作上下文为用户提供一些有效的警示框，比如说告诉用户操作成功、操作错误、提示或者警告等。如下图所示：

[](http://img.mukewang.com/5418f5120001a99e06760090.jpg)

## 2.1 警示框--默认警示框

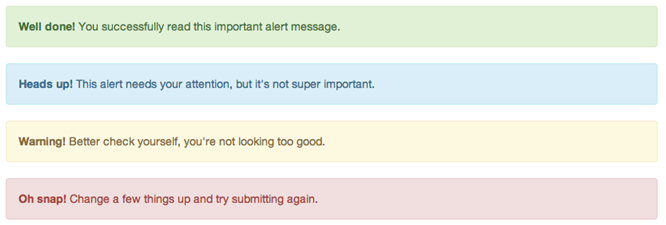
Bootstrap框架通过**“alert“样式来实现警示框效果**。在默认情况之下，提供了四种不同的警示框效果：

**1、成功警示框：**告诉用用户操作成功，**在“alert”样式基础上追加“alert-success”样式**，具体呈现的是背景、边框和文本都是绿色；

**2、信息警示框：**给用户提供提示信息，**在“alert”样式基础上追加“alert-info”样式**，具体呈现的是背景、边框和文本都是浅蓝色；

**3、警告警示框：**提示用户小心操作（提供警告信息），**在“alert”样式基础上追加“alert-warning”**样式，具体呈现的是背景、边框、文本都是浅黄色；

**4、错误警示框：**提示用户操作错误，**在“alert”样式基础上追加“alert-danger”样式**，具体呈现的是背景、边框和文本都是浅红色。

如下图示：[](http://img.mukewang.com/5418f5c400016e3006660225.jpg)

使用方法：

具体使用的时候，**可以在类名为“alert”的div容器里放置提示信息**。实现不同类型警示框，**只需要在“alert”基础上追加对应的类名**，如下：

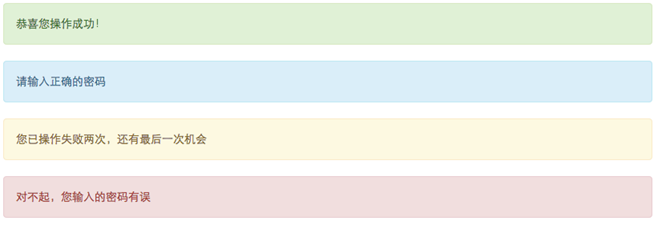
<div class="alert alert-success" role="alert">恭喜您操作成功！</div>

<div class="alert alert-info" role="alert">请输入正确的密码</div>

<div class="alert alert-warning" role="alert">您已操作失败两次，还有最后一次机会</div>

<div class="alert alert-danger" role="alert">对不起，您输入的密码有误</div>

运行效果如下：

[](http://img.mukewang.com/5418f6300001d5b206600225.jpg)

[](http://img.mukewang.com/5437857d00010f7004870305.jpg)

## 2.2 警示框--可关闭的警示框

大家在平时浏览网页的时候，会发现一些警示框带有关闭按钮，用户一点击关闭按钮就能自动关闭显示的警示框（也就是让警示框隐藏不显示）。在Bootstrap框架中的警示框也具有这样的功能。

使用方法：

只需要在默认的警示框里面添加一个关闭按钮。然后进行三个步骤：

**1、需要在基本警示框“alert”的基础上添加“alert-dismissable”样式。**

**2、在提示容器里面加上button，在button标签中加入class="close"类，实现警示框关闭按钮的样式。Button{&times;}**

**3、要确保关闭按钮元素上设置了自定义属性：“data-dismiss="alert"”（因为可关闭警示框需要借助于Javascript来检测该属性，从而控制警示框的关闭）。**

具体使用如下：

<div class="alert alert-success ****alert-dismissable****" role="alert">

<button ****class="close"**** type="button" ****data-dismiss="alert"****>&times;</button>

恭喜您操作成功！

</div>

运行效果如下：

[](http://img.mukewang.com/5418f90a0001127e06660238.jpg)

## 2.3 警示框--警示框的链接

有时候你可能想在警示框中加入链接地址，用来告诉用户跳到某一个地方或新的页面。而这个时候你又想让用户能明显的看出来这是链接地址。在Bootstrap框架中对警示框里的链接样式做了一个高亮显示处理。为不同类型的警示框内的链接进行了加粗处理，并且颜色相应加深。

实现方法：

Bootstrap框架是通过**给警示框加的链接添加一个名为“alert-link”的类名**，通过“alert-link”样式给链接提供高亮显示。

具体使用如下：

<div class="alert alert-success" role="alert">

<strong>Well done!</strong>

You successfully read

<a href="#" class="alert-link">this important alert message</a>

.

</div>

<div class="alert alert-info" role="alert">

<strong>Heads up!</strong>

This

<a href="#" class="alert-link">alert needs your attention</a>

, but it's not super important.

</div>

<div class="alert alert-warning" role="alert">

<strong>Warning!</strong>

Better check yourself, you're

<a href="#" class="alert-link">not looking too good</a>

.

</div>

<div class="alert alert-danger" role="alert">

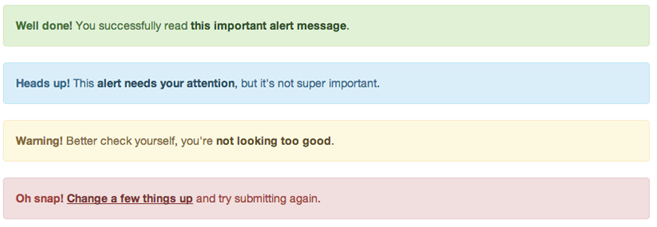
<strong>Oh snap!</strong>

<a href="#" class="alert-link">Change a few things up</a>

and try submitting again.

</div>

运行效果如下：

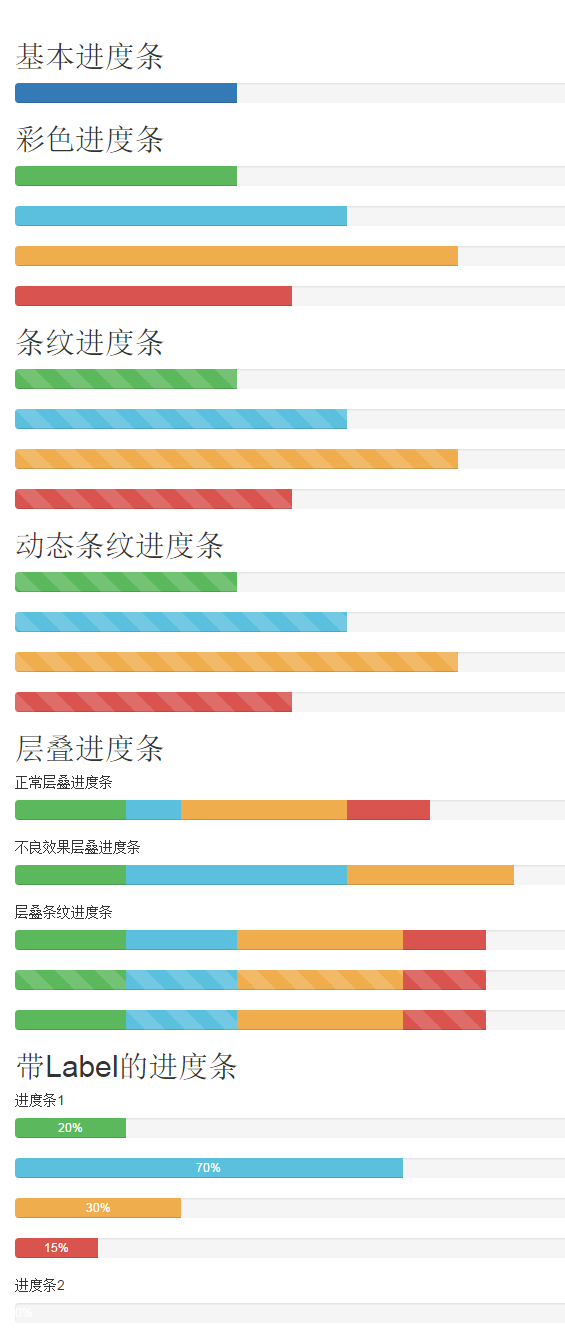
[](http://img.mukewang.com/5418fc470001ee6306570227.jpg)

# 3 进度条

在网页中，进度条的效果并不少见，比如一个评分系统，比如加载状态等。就如下图所示的一个评分系统，他就是一个简单的进度条效果：

[](http://img.mukewang.com/5418fe4e000106af05640152.jpg)

而且Bootstrap框架为大家提供多种样式风格的进度条，供大家使用，接下来我们将详细讲解这些进度条如何使用。



## 3.1 进度条--基本样式

Bootstrap框架中对于进度条提供了一个基本样式，一个100%宽度的背景色，然后个高亮的色表示完成进度。其实制作这样的进度条非常容易，一般是使用两个容器，外容器具有一定的宽度，并且设置一个背景颜色，他的子元素设置一个宽度，比如完成度是30%（也就是父容器的宽度比例值），同时给其设置一个高亮的背景色。

**使用方法：**

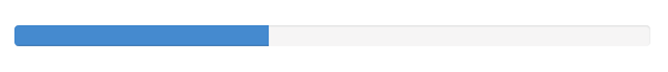
Bootstrap框架中也是按这样的方式实现的，他提供了两个容器，**外容器使用“progress”样式，子容器使用“progress-bar”样式。其中progress用来设置进度条的容器样式，而progress-bar用于限制进度条的进度**。使用方法非常的简单：

<div class="progress">

<div class="progress-bar" style="width:40%"></div>

</div>

运行效果如下：

[](http://img.mukewang.com/5418ff230001f9e106680072.jpg)

## 3.2 进度条--彩色进度条

Bootstrap框架中的进度条和警告信息框一样，为了能给用户一个更好的体验，也根据不同的状态配置了不同的进度条颜色。在此称为彩色进度条，其主要包括以下四种：

**☑ progress-bar-info：表示信息进度条，进度条颜色为蓝色**

**☑ progress-bar-success：表示成功进度条，进度条颜色为绿色**

**☑ progress-bar-warning：表示警告进度条，进度条颜色为黄色**

**☑ progress-bar-danger：表示错误进度条，进度条颜色为红色**

**使用方法：**

具体使用就非常简单了，只需要在基础的进度上增加对应的类名。如：

<div class="progress">

<div class="progress-bar progress-bar-success" style="width:40%"></div>

</div>

<div class="progress">

<div class="progress-bar progress-bar-info" style="width:60%"></div>

</div>

<div class="progress">

<div class="progress-bar progress-bar-warning" style="width:80%"></div>

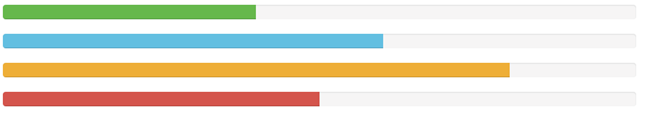
</div>

<div class="progress">

<div class="progress-bar progress-bar-danger" style="width:50%"></div>

</div>

运行效果如下：

[](http://img.mukewang.com/5419029300010f1806460117.jpg)

## 3.2 进度条--动态条纹进度条

**使用方法：**

**在进度条“progress progress-striped”两个类的基础上再加入“active”类名**。如下代码：

<div class="progress **progress-striped** ****active****">

<div class="progress-bar progress-bar-success" style="width:40%"></div>

</div>

**实现原理：**

为了让条纹进度条动起来，Bootstrap框架还提供了一种动态条纹进度条。其实现原理主要通过CSS3的animation来完成。首先通过@keyframes创建了一个progress-bar-stripes的动画，这个动画主要做了一件事，就是改变背景图像的位置，也就是background-position的值。因为条纹进度条是通过CSS3的线性渐变来制作的，而linear-gradient实现的正是对应背景中的背景图片。

@keyframes仅仅是创建了一个动画效果，如果要让进度条真正的动起来，我们需要通过一定的方式调用@keyframes创建的动画“progress-bar-stripes”，并且通过一个事件触发动画生效。在Bootstrap框架中，通过给进度条容器“progress”添加一个类名“active”，并让文档加载完成就触“progress-bar-stripes”动画生效。

<div class="progress progress-striped active">

<div class="progress-bar progress-bar-success" style="width:40%"></div>

</div>

<div class="progress progress-striped active">

<div class="progress-bar progress-bar-info" style="width:60%"></div>

</div>

<div class="progress progress-striped active">

<div class="progress-bar progress-bar-warning" style="width:80%"></div>

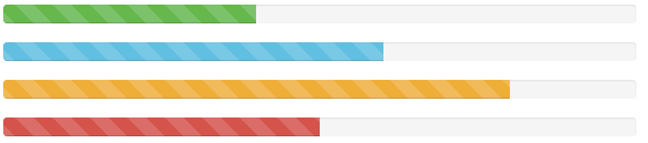
</div>

<div class="progress progress-striped active">

<div class="progress-bar progress-bar-danger" style="width:50%"></div>

</div>

运行效果如下：

[](http://img.mukewang.com/5419229f00011bbe06480143.jpg)

**特别注意：要让条纹进度条动起来，就需要让“progress-striped”和“active”同时运用，不然条纹进度条是不具备动效效果。**

## 3.3 进度条--层叠进度条

Bootstrap框架除了提供上述几种进度条之外，还提供了一种层叠进度条，层叠进度条，可以将不同状态的进度条放置在一起，按水平方式排列。具体使用如下：

<div class="progress">

<div class="progress-bar progress-bar-success" style="width:20%"></div>

<div class="progress-bar progress-bar-info" style="width:10%"></div>

<div class="progress-bar progress-bar-warning" style="width:30%"></div>

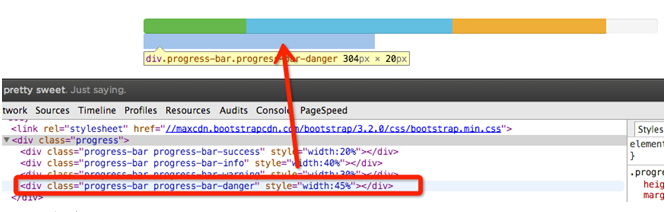
<div class="progress-bar progress-bar-danger" style="width:15%"></div>

</div>

运行效果如下：

[](http://img.mukewang.com/5419242a0001b6cc06660074.jpg)

或许你会感到疑问，没有为层叠进度条添加额外的样式代码，怎么就正常了呢？可以回过头来看基本进度条那部分，不难发现，在“progress-bar”上有一个左浮动的样式。也就是这个样式，在不增加任何样式代码就能实现上例的层叠效果。当然有一点需要注意，层叠进度条宽度之和不能大于100%，大于100%就会造成下面的不良效果：

[](http://img.mukewang.com/541924530001ed6e06650212.jpg)

除了层叠彩色进度条之外，还可以层叠条纹进度条，或者说条纹进度条和彩色进度条混合层叠，仅需要在“progress”容器中添加对应的进度条，同样要注意，层叠的进度条之和不能大于100%。来简单的看一个示例：

<div class="progress">

<div class="progress-bar progress-bar-success" style="width:20%"></div>

<div class="progress-bar progress-bar-info" style="width:20%"></div>

<div class="progress-bar progress-bar-warning" style="width:30%"></div>

<div class="progress-bar progress-bar-danger" style="width:15%"></div>

</div>

<div class="progress">

<div class="progress-bar progress-bar-success progress-bar-striped" style="width:20%"></div>

<div class="progress-bar progress-bar-info progress-bar-striped" style="width:20%"></div>

<div class="progress-bar progress-bar-striped progress-bar-warning" style="width:30%"></div>

<div class="progress-bar progress-bar-danger progress-bar-striped" style="width:15%"></div>

</div>

<div class="progress">

<div class="progress-bar progress-bar-success" style="width:20%"></div>

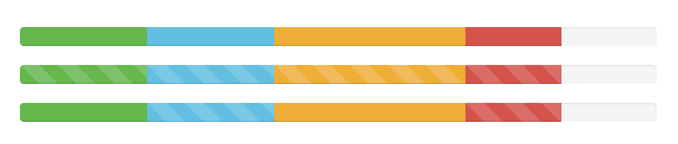
<div class="progress-bar progress-bar-info progress-bar-striped" style="width:20%"></div>

<div class="progress-bar progress-bar-warning" style="width:30%"></div>

<div class="progress-bar progress-bar-danger progress-bar-striped" style="width:15%"></div>

</div>

运行效果如下：

[](http://img.mukewang.com/541924ce0001a3b606780144.jpg)

## 3.4 进度条--带Label的进度条

上面介绍的各种进度条，都仅仅是通过颜色进度向用户传递进度值。但实际中，有很多时候是需要在进度条中直接用相关的数值向用户传递完成的进度值，在Bootstrap就为大家考虑了这种使用场景。

**实现方法：**

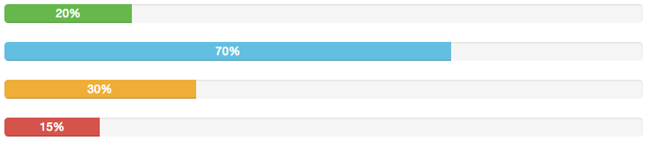
只需要在进度条中添加你需要的值，如：

<div class="progress">

<div class="progress-bar progress-bar-success" style="width:20%">20%</div>

</div>

运行效果如下：

[](http://img.mukewang.com/541928420001baea06610148.jpg)

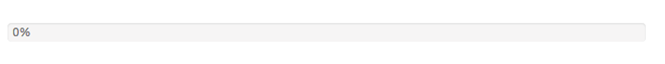
还有一种特殊情形，当进度条处于开始位置，也就是进度条的值为0%时，内容是否会撑开一定的宽度，让进度条具有颜色呢？如果是，这不是我们需要的效果，如果不是，又是怎么实现的呢？我们先来看一个这样的示例：

<div class="progress">

<div class="progress-bar" role="progressbar" aria-valuenow="0" aria-valuemin="0" aria-valuemax="100">0%</div>

</div>

运行效果如下：

[](http://img.mukewang.com/541928ab0001b41506650064.jpg)

# 4 媒体对象

在Web页面或者说移动页面制作中，常常看到这样的效果，左边居左（或居右），内容居右（或居左）排列，如下图所示：

[](http://img.mukewang.com/54192a4d00014b6a06590154.jpg)

我们常常把这样的效果称为媒体对象。可以说他是一种抽像的样式，可以用来构建不同类型的组件。这些组件都具有开篇所说的样式风格。那么在Bootstrap框架中特意将些部分提取出来做来一个组件介绍。

## 4.1 媒体对象--默认媒体对象

媒体对象一般是成组出现，而一组媒体对象常常包括以下几个部分：

  ☑ **媒体对像的容器**：常使用“media”类名表示，用来容纳媒体对象的所有内容

  ☑  **媒体对像的对象**：常使用“media-object”表示，就是媒体对象中的对象，常常是图片

  ☑  **媒体对象的主体**：常使用“media-body”表示，就是媒体对像中的主体内容，可以是任何元素，常常是图片侧边内容

  ☑  **媒体对象的标题**：常使用“media-heading”表示，就是用来描述对象的一个标题，此部分可选

如下图所示：



除了上面四个部分之外，在Bootstrap框架中还**常常使用“pull-left”或者“pull-right”来控制媒体对象中的对象浮动方式**。

在具体使用中如下所示：

<div class="container">

<div class="media">

<a href="http://www.badu.com" class="pull-left">

<img src="book.jpg" class="media-object"/>

</a>

<div class="media-body">

<h2 class="media-heading">HTML5 游戏开发实践指南</h2>

<p>

是2012年机械工业出版社出版的图书。原书名：Learning HTML5 Game Programming: A Hands-on Guide to Building Online Games Using Canvas, SVG, and WebGL 作者： (美)Williams,J.L. [作译者介绍] 丛书名： 华章专业开发者丛书 上架时间：2012-5-16 出版日期：2012 年5月

</p>

</div>

</div>

</div>

运行效果如下：



## 4.2 媒体对象--媒体对象列表

媒体对象的嵌套仅是媒体对象中一个简单应用效果之一，在很多时候，我们还会碰到一个列表，每个列表项都和媒体对象长得差不多，同样用评论系统来说事：

[](http://img.mukewang.com/541930820001e33e06580376.jpg)

使用方法：

针对上图的媒体对象列表效果，Bootstrap框架提供了一个列表展示的效果，**在写结构的时候可以使用ul，并且在ul上添加类名“media-list”，而在li上使用“media”**，示例代码如下：

<ul class="****media-list****">

<li class="****media****">

<a class="pull-left" href="#">

<img class="media-object" src=" " alt="...">

</a>

<div class="media-body">

<h4 class="media-heading">Media Header</h4>

<div>…</div>

</div>

</li>

<li class="****media****">…</li>

<li class="****media****">…</li>

</ul>

# 5 列表组

列表组是Bootstrap框架新增的一个组件，可以用来制作列表清单、垂直导航等效果，也可以配合其他的组件制作出更漂亮的组件。

## 5.1 列表组--基础列表组

基础列表组，看上去就是去掉了列表符号的列表项，并且配上一些特定的样式。在Bootstrap框架中的基础列表组主要包括两个部分：

**☑  list-group：列表组容器，常用的是ul元素，当然也可以是ol或者div元素**

**☑  list-group-item：列表项，常用的是li元素，当然也可以是div元素**

**来看一个简单的示例：**

<ul class="****list-group****">

<li class="****list-group-item****">揭开CSS3的面纱</li>

<li class="****list-group-item****">CSS3选择器</li>

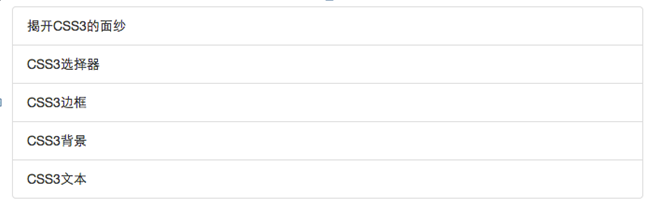
<li class="****list-group-item****">CSS3边框</li>

<li class="****list-group-item****">CSS3背景</li>

<li class="****list-group-item****">CSS3文本</li>

</ul>

运行效果如下：

[](http://img.mukewang.com/541938050001335e06550202.jpg)

## 5.2 列表组--带徽章的列表组

带徽章的列表组其实就是将Bootstrap框架中的徽章组件和基础列表组结合在一起的一个效果。具体做法很简单，只需要在“list-group-item”中添加徽章组件“badge”：

<ul class="list-group">

<li class="list-group-item">

****<span class="badge">13</span>****揭开CSS3的面

</li>

<li class="list-group-item">

****<span class="badge">456</span>****CSS3选择器

</li>

<li class="list-group-item">

****<span class="badge">892</span>****CSS3边框

</li>

<li class="list-group-item">

****<span class="badge">90</span>****CSS3背景

</li>

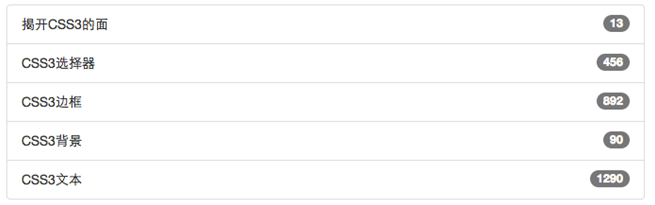
<li class="list-group-item">

****<span class="badge">1290</span>****CSS3文本

</li>

</ul>

运行效果如下：

[](http://img.mukewang.com/541939cb000111e006550205.jpg)

实现原理：

实现效果非常简单，就是给徽章设置了一个右浮动，当然如果有两个徽章同时在一个列表项中出现时。

## 5.3 列表组--带链接的列表组

带链接的列表组，其实就是每个列表项都具有链接效果。大家可能最初想到的就是在基础列表组的基础上，给列表项的文本添加链接：

<ul class="list-group">

<li class="list-group-item">

<a href="##">揭开CSS3的面</a>

</li>

<li class="list-group-item">

<a href="##">CSS3选择器</a>

</li>

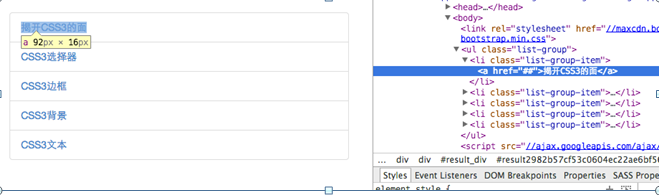
...

</ul>

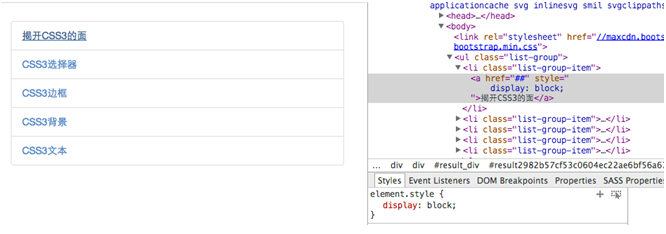
运行效果如下：

[](http://img.mukewang.com/54193e2b0001e6e606470202.jpg)

**这样做有一个不足之处，就是链接的点击区域只在文本上有效：**

[](http://img.mukewang.com/54193e7300017f9406590195.jpg)

但很多时候，都希望在列表项的任何区域都具备可点击。这个时候就需要在**链接标签上增加额外的样式：“display:block”；**

[](http://img.mukewang.com/54193eba00018ae306660234.jpg)

虽然这样能解决问题，达到需求。但在Bootstrap框架中，还是采用了另一种实现方式。就是**将ul.list-group使用div.list-group来替换，而li.list-group-item直接用a.list-group-item来替换**。这样就可以达到需要的效果：

<div class="list-group">

<a href="##" class="list-group-item">图解CSS3</a>

<a href="##" class="list-group-item"><span class="badge">220</span>Sass教程</a>

<a href="##" class="list-group-item">玩转Bootstrap</a>

</div>

[](http://img.mukewang.com/54193f4d0001e76b06570137.jpg)

## 5.4 列表组--自定义列表组

Bootstrap框加在链接列表组的基础上新增了两个样式：

**☑  list-group-item-heading：用来定义列表项头部样式**

**☑  list-group-item-text：用来定义列表项主要内容**

这两个样式最大的作用就是用来帮助开发者可以自定义列表项里的内容，如下面的示例：

<div class="list-group">

<a href="##" class="list-group-item">

<h4 class="****list-group-item-heading****">图解CSS3</h4>

<p class="****list-group-item-text****">...</p>

</a>

<a href="##" class="list-group-item">

<h4 class="****list-group-item-heading****">Sass中国</h4>

<p class="****list-group-item-text****">...</p>

</a>

</div>

运行效果如下：

[](http://img.mukewang.com/541941c700010e7a06490147.jpg)

## 5.5 列表组--列表项的状态设置

Bootstrap框架也给组合列表项提供了状态效果，特别是链接列表组。比如常见状态和禁用状态等。实现方法和前面介绍的组件类似，在列表组中只需要在对应的列表项中添加类名：

**☑  active：表示当前状态**

**☑  disabled：表示禁用状态**

来看个示例：

<div class="list-group">

<a href="##" class="list-group-item active"><span class="badge">5902</span>图解CSS3</a>

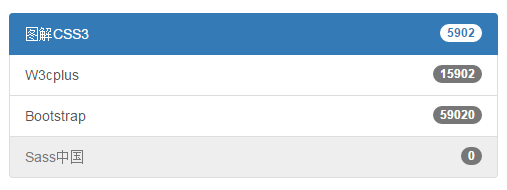
<a href="##" class="list-group-item"><span class="badge">15902</span>W3cplus</a>

<a href="##" class="list-group-item"><span class="badge">59020</span>Bootstrap</a>

<a href="##" class="list-group-item disabled"><span class="badge">0</span>Sass中国</a>

</div>

运行效果如下：



## 5.6 列表组--多彩列表组

列表组组件和警告组件一样，Bootstrap为不同的状态提供了不同的背景颜色和文本色，可以使用这几个类名定义不同背景色的列表项。

  ☑  list-group-item-success：成功，背景色绿色

  ☑  list-group-item-info：信息，背景色蓝色

  ☑  list-group-item-warning：警告，背景色为黄色

  ☑  list-group-item-danger：错误，背景色为红色

如果你想给列表项添加什么背景色，**只需要在“list-group-item”基础上增加对应的类名**：

<div class="list-group">

<a href="##" class="list-group-item active"><span class="badge">5902</span>图解CSS3</a>

<a href="##" class="list-group-item list-group-item-success"><span class="badge">15902</span>W3cplus</a>

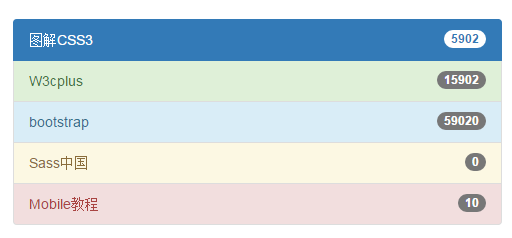
<a href="##" class="list-group-item list-group-item-info"><span class="badge">59020</span>bootstrap</a>

<a href="##" class="list-group-item list-group-item-warning"><span class="badge">0</span>Sass中国</a>

<a href="##" class="list-group-item list-group-item-danger"><span class="badge">10</span>Mobile教程</a>

</div>

运行效果如下：



# 6 面板

面板（Panels）是Bootstrap框架新增的一个组件，其主要作用就是用来处理一些其他组件无法完成的功能。同样在不同的版本中具有不同的源码：

## 6.1 面板--基础面板

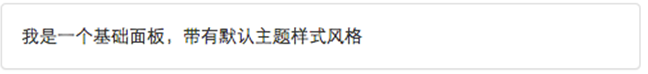
基础面板非常简单，就是一个**div容器运用了“panel”样式**，产生一个具有边框的文本显示块。**由于“panel”不控制主题颜色，所以在“panel”的基础上增加一个控制颜色的主题“panel-default”**，另外在里面添加了一个“div.panel-body” 来放置面板主体内容：

<div class="****panel panel-default****">

<div class="****panel-body****">我是一个基础面板，带有默认主题样式风格</div>

</div>

运行效果如下：

[](http://img.mukewang.com/54194ead00016f8806480074.jpg)

## 6.2 面板--带有头和尾的面板

基础面板看上去太简单了，Bootstrap为了丰富面板的功能，特意为面板增加“**面板头部**”和“**页面尾部**”的效果：

**☑  panel-heading：用来设置面板头部样式**

**☑ panel-footer：用来设置面板尾部样式**

<div class="panel panel-default">

<div class="panel-heading">图解CSS3</div>

<div class="panel-body">…</div>

<div class="panel-footer">作者：大漠</div>

</div>

运行效果如下：

[](http://img.mukewang.com/5419500b00017cc706440219.jpg)

## 6.3 面板--彩色面板

在基础面板一节中了解到，panel样式并没有对主题进行样式设置，而主题样式是通过panel-default来设置。在Bootstrap框架中面板组件除了默认的主题样式之外，还包括以下几种主题样式，构成了一个彩色面板：

**☑  panel-primary：重点蓝**

**☑  panel-success：成功绿**

**☑ panel-info:信息蓝**

**☑ panel-warning：警告黄**

**☑ panel-danger：危险红**

使用方法就很简单了，只需要在panel的类名基础上增加自己需要的类名：

<div class="panel panel-default">

<div class="panel-heading">图解CSS3</div>

<div class="panel-body">…</div>

<div class="panel-footer">作者：大漠</div>

</div>

<div class="panel panel-primary">…</div>

<div class="panel panel-success">…</div>

<div class="panel panel-info">…</div>

<div class="panel panel-warning">…</div>

<div class="panel panel-danger">…</div>

[](http://img.mukewang.com/541951700001139606510546.jpg)

## 6.4 面板--面板中嵌套列表组

学习如何在面板中放置列表组，我们简单的来看一个示例：

<div class="panel panel-default">

<div class="panel-heading">图解CSS3</div>

<div class="panel-body">

<p>详细讲解了选择器、边框、背景、文本、颜色、盒模型、伸缩布局盒模型、多列布局、渐变、过渡、动画、媒体、响应Web设计、Web字体等主题下涵盖的所有CSS3新特性

</p>

<ul class="list-group">

<li class="list-group-item">我是列表项</li>

<li class="list-group-item">我是列表项</li>

<li class="list-group-item">我是列表项</li>

</ul>

</div>

<div class="panel-footer">作者：大漠</div>

</div>

运行效果如下：

[](http://img.mukewang.com/541957920001280d06600435.jpg)

**优化代码：**

如果你觉得这样有间距不好看，你完全可以把列表组提取出来：

<div class="panel panel-default">

<div class="panel-heading">图解CSS3</div>

<div class="panel-body">…</div>

<ul class="list-group">

<li class="list-group-item">我是列表项</li>

<li class="list-group-item">我是列表项</li>

<li class="list-group-item">我是列表项</li>

</ul>

<div class="panel-footer">作者：大漠</div>

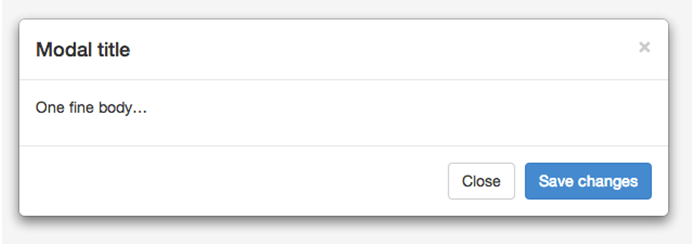
</div>

运行效果如下：

[](http://img.mukewang.com/541957ed000127f106440360.jpg)

# 7 模态弹出框（Modals）

在 Bootstrap 框架中把模态弹出框统一称为 Modal 。这种弹出框效果在大多数 Web 网站的交互中都可见。比如点击一个按钮弹出一个框，弹出的框可能是一段文件描述，也可能带有按钮操作，也有可能弹出的是一张图片。如下图所示：



**实现原理解析：**

bootstrap中的“模态弹出框”有以下几个特点：

**1、**模态弹出窗是固定在浏览器中的。

**2、**单击右侧全屏按钮，在全屏状态下，模态弹出窗宽度是自适应的，而且modal-dialog水平居中。

**3、**当浏览器视窗大于768px时，模态弹出窗的宽度为600px。