

Bootstrap 布局组件

Bootstrap 字体图标(Glyphicons)

本章将讲解字体图标(Glyphicons)，并通过一些实例了解它的使用。Bootstrap 捆绑了 200 多种字体格式的字形。首先让我们先来理解一下什么是字体图标。

什么是字体图标？

字体图标是在 Web 项目中使用的图标字体。虽然，[Glyphicons Halflings](#) 需要商业许可，但是您可以通过基于项目的 Bootstrap 来免费使用这些图标。

为了表示对图标作者的感谢，希望您在使用时加上 GLYPHICONS 网站的链接。

获取字体图标

我们已经在 [环境安装](#) 章节下载了 Bootstrap 3.x 版本，并理解了它的目录结构。在 *fonts* 文件夹内可以找到字体图标，它包含了下列这些文件：

- glyphsicons-halflings-regular.eot
- glyphsicons-halflings-regular.svg
- glyphsicons-halflings-regular.ttf
- glyphsicons-halflings-regular.woff

相关的 CSS 规则写在 *dist* 文件夹内的 *css* 文件夹内的 *bootstrap.css* 和 *bootstrap-min.css* 文件上。

字体图标列表

[点击这里，查看可用的字体图标列表。](#)

CSS 规则解释

下面的 CSS 规则构成 glyphicon class。

```
@font-face {
  font-family: 'Glyphicons Halflings';
  src: url('../fonts/glyphicons-halflings-regular.eot');
  src:          url('../fonts/glyphicons-halflings-regular.eot?#iefix')          format('embedded-opentype'),
url('../fonts/glyphicons-halflings-regular.woff')    format('woff'),    url('../fonts/glyphicons-halflings-regular.ttf')
format('truetype'), url('../fonts/glyphicons-halflings-regular.svg#glyphicons_halfllingsregular') format('svg');
}
```

```
.glyphicon {
  position: relative;
  top: 1px;
  display: inline-block;
  font-family: 'Glyphicons Halflings';
  -webkit-font-smoothing: antialiased;
  font-style: normal;
  font-weight: normal;
  line-height: 1;
```

```
-moz-osx-font-smoothing: grayscale;
}
```

所以 font-face 规则实际上是在找到 `glyphicons` 地方声明 `font-family` 和位置。

`.glyphicon` class 声明一个从顶部偏移 1px 的相对位置，呈现为 `inline-block`，声明字体，规定 `font-style` 和 `font-weight` 为 `normal`，设置行高为 1。除此之外，使用 `-webkit-font-smoothing: antialiased` 和 `-moz-osx-font-smoothing: grayscale;` 获得跨浏览器的一致性。

然后，这里的

```
.glyphicon:empty {
  width: 1em;
}
```

是空的 `glyphicon`。

这里有 200 个 class，每个 class 针对一个图标。这些 class 的常见格式如下：

```
.glyphicon-keyword:before {
  content: "hexvalue";
}
```

比如，使用的 `user` 图标，它的 class 如下：

```
.glyphicon-user:before {
  content: "\e008";
}
```

用法

如需使用图标，只需要简单地使用下面的代码即可。请在图标和文本之间保留适当的空间。

```
<span class="glyphicon glyphicon-search"></span>
```

下面的实例演示了如何使用字体图标：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 如何使用字体图标</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<p>
  <button type="button" class="btn btn-default">
    <span class="glyphicon glyphicon-sort-by-attributes"></span>
  </button>
  <button type="button" class="btn btn-default">
    <span class="glyphicon glyphicon-sort-by-attributes-alt"></span>
  </button>
  <button type="button" class="btn btn-default">
    <span class="glyphicon glyphicon-sort-by-order"></span>
  </button>
```

```

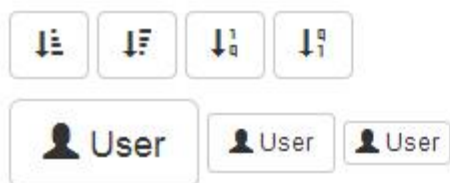
<button type="button" class="btn btn-default">
  <span class="glyphicon glyphicon-sort-by-order-alt"></span>
</button>
</p>
<button type="button" class="btn btn-default btn-lg">
  <span class="glyphicon glyphicon-user"></span> User
</button>
<button type="button" class="btn btn-default btn-sm">
  <span class="glyphicon glyphicon-user"></span> User
</button>
<button type="button" class="btn btn-default btn-xs">
  <span class="glyphicon glyphicon-user"></span> User
</button>

```

</body>

</html>

结果如下所示:



带有导航栏的字体图标

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>导航栏的字体图标</title>
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<!-- Bootstrap -->
```

```
<link href="http://apps.bdimg.com/libs/bootstrap/3.2.0/css/bootstrap.min.css" rel="stylesheet">
```

```
<style>
```

```
body {
```

```
padding-top: 50px;
```

```
padding-left: 50px;
```

```
}
```

```
</style>
```

```
<!--[if lt IE 9]>
```

```
<script src="http://apps.bdimg.com/libs/html5shiv/3.7/html5shiv.min.js"></script>
```

```
<![endif]-->
```

```
</head>
```

```
<body>
```

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

```

<div class="container">
  <div class="navbar-header">
    <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
      <span class="sr-only">Toggle navigation</span>
      <span class="icon-bar"></span>
      <span class="icon-bar"></span>
      <span class="icon-bar"></span>
    </button>
    <a class="navbar-brand" href="#">Project name</a>
  </div>
  <div class="collapse navbar-collapse">
    <ul class="nav navbar-nav">
      <li class="active"><a href="#"><span class="glyphicon glyphicon-home">Home</span></a></li>
      <li><a href="#shop"><span class="glyphicon glyphicon-shopping-cart">Shop</span></a></li>
      <li><a href="#support"><span class="glyphicon glyphicon-headphones">Support</span></a></li>
    </ul>
  </div><!-- /.nav-collapse -->
</div><!-- /.container -->
</div>
<!-- jQuery (Bootstrap 插件需要引入) -->
<script src="http://apps.bdimg.com/libs/jquery/2.1.1/jquery.min.js"></script>
<!-- 包含了所有编译插件 -->
<script src="http://apps.bdimg.com/libs/bootstrap/3.2.0/js/bootstrap.min.js"></script>
</body>
</html>

```

定制字体图标

我们已经看到如何使用字体图标，接下来我们看看如何定制字体图标。

我们将以上面的实例开始，并通过改变字体尺寸、颜色和应用文本阴影来进行定制图标。

下面是开始的代码：

```

<button type="button" class="btn btn-primary btn-lg">
  <span class="glyphicon glyphicon-user"></span> User
</button>

```

效果如下所示：



定制字体尺寸

通过增加或减少图标的字体尺寸，您可以让图标看起来更大或更小。

```

<button type="button" class="btn btn-primary btn-lg" style="font-size: 60px">
  <span class="glyphicon glyphicon-user"></span> User
</button>

```



定制字体颜色

```
<button type="button" class="btn btn-primary btn-lg" style="color: rgb(212, 106, 64);">
  <span class="glyphicon glyphicon-user"></span> User
</button>
```



应用文本阴影

```
<button type="button" class="btn btn-primary btn-lg" style="text-shadow: black 5px 3px 3px;">
  <span class="glyphicon glyphicon-user"></span> User
</button>
```



Bootstrap 下拉菜单 (Dropdowns)

本章将重点介绍 Bootstrap 下拉菜单。下拉菜单是可切换的，是以列表格式显示链接的上下文菜单。这可以通过与 [下拉菜单 \(Dropdown\) JavaScript 插件](#) 的互动来实现。

如需使用下列菜单，只需要在 class **.dropdown** 内加上下拉菜单即可。下面的实例演示了基本的下拉菜单：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 下拉菜单 (Dropdowns) </title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="dropdown">
  <button type="button" class="btn dropdown-toggle" id="dropdownMenu1"
    data-toggle="dropdown">
    主题
    <span class="caret"></span>
  </button>
  <ul class="dropdown-menu" role="menu" aria-labelledby="dropdownMenu1">
    <li role="presentation">
      <a role="menuitem" tabindex="-1" href="#">Java</a>
    </li>
    <li role="presentation">
      <a role="menuitem" tabindex="-1" href="#">数据挖掘</a>
    </li>
    <li role="presentation">
      <a role="menuitem" tabindex="-1" href="#">
        数据通信/网络
      </a>
    </li>
  </ul>
</div>
```

```

        </a>
    </li>
    <li role="presentation" class="divider"></li>
    <li role="presentation">
        <a role="menuitem" tabindex="-1" href="#">分离的链接</a>
    </li>
</ul>
</div>

```

```
</body>
```

```
</html>
```

结果如下所示：



选项

对齐

通过向 **.dropdown-menu** 添加 class **.pull-right** 来向右对齐下拉菜单。下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 向右对齐下拉菜单</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="dropdown">
    <button type="button" class="btn dropdown-toggle" id="dropdownMenu1"
        data-toggle="dropdown">主题
        <span class="caret"></span>
    </button>
    <ul class="dropdown-menu pull-right" role="menu"
        aria-labelledby="dropdownMenu1">
        <li role="presentation">

```

```

        <a role="menuitem" tabindex="-1" href="#">Java</a>
    </li>
    <li role="presentation">
        <a role="menuitem" tabindex="-1" href="#">数据挖掘</a>
    </li>
    <li role="presentation">
        <a role="menuitem" tabindex="-1" href="#">
            数据通信/网络
        </a>
    </li>
    <li role="presentation" class="divider"></li>
    <li role="presentation">
        <a role="menuitem" tabindex="-1" href="#">分离的连接</a>
    </li>
</ul>
</div>

```

</body>

</html>

结果如下所示:



标题

您可以使用 class **dropdown-header** 向下拉菜单的标签区域添加标题。下面的实例演示了这点:

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 下拉菜单标题</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="dropdown">

```

```

<button type="button" class="btn dropdown-toggle" id="dropdownMenu1"
  data-toggle="dropdown">
  主题
  <span class="caret"></span>
</button>
<ul class="dropdown-menu" role="menu" aria-labelledby="dropdownMenu1">
  <li role="presentation" class="dropdown-header">下拉菜单标题</li>
  <li role="presentation" >
    <a role="menuitem" tabindex="-1" href="#">Java</a>
  </li>
  <li role="presentation">
    <a role="menuitem" tabindex="-1" href="#">数据挖掘</a>
  </li>
  <li role="presentation">
    <a role="menuitem" tabindex="-1" href="#">
      数据通信/网络
    </a>
  </li>
  <li role="presentation" class="divider"></li>
  <li role="presentation" class="dropdown-header">下拉菜单标题</li>
  <li role="presentation">
    <a role="menuitem" tabindex="-1" href="#">分离的链接</a>
  </li>
</ul>
</div>

```

</body>

</html>

结果如下所示:



Bootstrap 按钮组

按钮组允许多个按钮被堆叠在同一行上。当你想要把按钮对齐在一起时，这就显得非常有用。您可以通过 Bootstrap 按钮（Button） 插件 添加可选的 JavaScript 单选框和复选框样式行为。

下面的表格总结了 Bootstrap 提供的使用按钮组的一些重要的 class：

Class	描述	代码示例
.btn-group	该 class 用于形成基本的按钮组。在 .btn-group 中放置一系列带有 class .btn 的按钮。	<pre><div class="btn-group"> <button type="button" class="btn btn-default">Button1</button> <button type="button" class="btn btn-default">Button2</button> </div></pre>
.btn-toolbar	该 class 有助于把几组 <code><div class="btn-group"></code> 结合到一个 <code><div class="btn-toolbar"></code> 中，一般获得更复杂的组件。	<pre><div class="btn-toolbar" role="toolbar"> <div class="btn-group">...</div> <div class="btn-group">...</div> </div></pre>
.btn-group-lg, .btn-group-sm, .btn-group-xs	这些 class 可应用到整个按钮组的大小调整，而不需要对每个按钮进行大小调整。	<pre><div class="btn-group btn-group-lg">...</div> <div class="btn-group btn-group-sm">...</div> <div class="btn-group btn-group-xs">...</div></pre>
.btn-group-vertical	该 class 让一组按钮垂直堆叠显示，而不是水平堆叠显示。	<pre><div class="btn-group-vertical"> ... </div></pre>

基本的按钮组

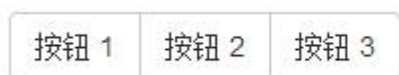
下面的实例演示了上面表格中讨论到的 class **.btn-group** 的使用：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 基本的按钮组</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="btn-group">
  <button type="button" class="btn btn-default">按钮 1</button>
  <button type="button" class="btn btn-default">按钮 2</button>
  <button type="button" class="btn btn-default">按钮 3</button>
</div>

</body>
</html>
```

结果如下所示：



按钮工具栏

下面的实例演示了上面表格中讨论到的 class **.btn-toolbar** 的使用：

```
<!DOCTYPE html>
<html>
<head>
```

```

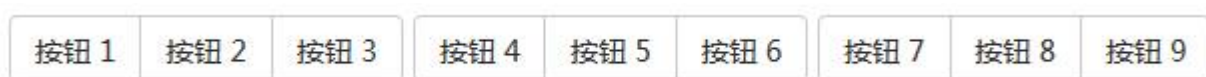
<title>Bootstrap 实例 - 按钮工具栏</title>
<link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<script src="/scripts/jquery.min.js"></script>
<script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="btn-toolbar" role="toolbar">
  <div class="btn-group">
    <button type="button" class="btn btn-default">按钮 1</button>
    <button type="button" class="btn btn-default">按钮 2</button>
    <button type="button" class="btn btn-default">按钮 3</button>
  </div>
  <div class="btn-group">
    <button type="button" class="btn btn-default">按钮 4</button>
    <button type="button" class="btn btn-default">按钮 5</button>
    <button type="button" class="btn btn-default">按钮 6</button>
  </div>
  <div class="btn-group">
    <button type="button" class="btn btn-default">按钮 7</button>
    <button type="button" class="btn btn-default">按钮 8</button>
    <button type="button" class="btn btn-default">按钮 9</button>
  </div>
</div>

</body>
</html>

```

结果如下所示：



按钮的大小

下面的实例演示了上面表格中讨论到的 class **.btn-group-*** 的使用：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 按钮组的大小</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

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

```

```

<button type="button" class="btn btn-default">按钮 1</button>
<button type="button" class="btn btn-default">按钮 2</button>
<button type="button" class="btn btn-default">按钮 3</button>
</div>
<div class="btn-group btn-group-sm">
<button type="button" class="btn btn-default">按钮 4</button>
<button type="button" class="btn btn-default">按钮 5</button>
<button type="button" class="btn btn-default">按钮 6</button>
</div>
<div class="btn-group btn-group-xs">
<button type="button" class="btn btn-default">按钮 7</button>
<button type="button" class="btn btn-default">按钮 8</button>
<button type="button" class="btn btn-default">按钮 9</button>
</div>

</body>
</html>

```

结果如下所示：



嵌套

您可以在一个按钮组内嵌套另一个按钮组，即，在一个 **.btn-group** 内嵌套另一个 **.btn-group**。当您向下拉菜单与一系列按钮组合使用时，就会用到这个。

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 嵌套的按钮组</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="btn-group">
  <button type="button" class="btn btn-default">按钮 1</button>
  <button type="button" class="btn btn-default">按钮 2</button>

  <div class="btn-group">
    <button type="button" class="btn btn-default dropdown-toggle"
      data-toggle="dropdown">
      下列
      <span class="caret"></span>
    </button>

```

```

        <ul class="dropdown-menu">
            <li><a href="#">下拉链接 1</a></li>
            <li><a href="#">下拉链接 2</a></li>
        </ul>
    </div>
</div>

</body>
</html>

```

结果如下所示：



垂直的按钮组

下面的实例演示了上面表格中讨论到的 class **.btn-group-vertical** 的使用：

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 垂直的按钮组</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="btn-group-vertical">
    <button type="button" class="btn btn-default">按钮 1</button>
    <button type="button" class="btn btn-default">按钮 2</button>

    <div class="btn-group-vertical">
        <button type="button" class="btn btn-default dropdown-toggle"
            data-toggle="dropdown">
            下拉
            <span class="caret"></span>
        </button>
        <ul class="dropdown-menu">
            <li><a href="#">下拉链接 1</a></li>
            <li><a href="#">下拉链接 2</a></li>
        </ul>
    </div>
    </div>
</div>

```

```
</body>
```

```
</html>
```

结果如下所示:



Bootstrap 按钮下拉菜单

本章将讲解如何使用 Bootstrap class 向按钮添加下拉菜单。如需向按钮添加下拉菜单，只需要简单地在在在一个 **.btn-group** 中放置按钮和下拉菜单即可。您也可以使用 `` 来指示按钮作为下拉菜单。

下面的实例演示了一个基本的简单的按钮下拉菜单:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Bootstrap 实例 - 基本的按钮下拉菜单</title>
```

```
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```
  <script src="/scripts/jquery.min.js"></script>
```

```
  <script src="/bootstrap/js/bootstrap.min.js"></script>
```

```
</head>
```

```
<body>
```

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

```
  <button type="button" class="btn btn-default dropdown-toggle"
```

```
    data-toggle="dropdown">
```

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

```
  </button>
```

```
  <ul class="dropdown-menu" role="menu">
```

```
    <li><a href="#">功能</a></li>
```

```
    <li><a href="#">另一个功能</a></li>
```

```
    <li><a href="#">其他</a></li>
```

```
    <li class="divider"></li>
```

```
    <li><a href="#">分离的链接</a></li>
```

```
  </ul>
```

```
</div>
```

```

<div class="btn-group">
  <button type="button" class="btn btn-primary dropdown-toggle"
    data-toggle="dropdown">
    原始 <span class="caret"></span>
  </button>
  <ul class="dropdown-menu" role="menu">
    <li><a href="#">功能</a></li>
    <li><a href="#">另一个功能</a></li>
    <li><a href="#">其他</a></li>
    <li class="divider"></li>
    <li><a href="#">分离的链接</a></li>
  </ul>
</div>

```

</body>

</html>

结果如下所示:



分割的按钮下拉菜单

分割的按钮下拉菜单使用与下拉菜单按钮大致相同的样式，但是对下拉菜单添加了原始的功能。分割按钮的左边是原始的功能，右边是显示下拉菜单的切换。

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 分割的按钮下拉菜单</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

```

```

<div class="btn-group">
  <button type="button" class="btn btn-default">默认</button>
  <button type="button" class="btn btn-default dropdown-toggle"
    data-toggle="dropdown">
    <span class="caret"></span>
    <span class="sr-only">切换下拉菜单</span>
  </button>
  <ul class="dropdown-menu" role="menu">
    <li><a href="#">功能</a></li>
    <li><a href="#">另一个功能</a></li>
    <li><a href="#">其他</a></li>
  </ul>

```

```

        <li class="divider"></li>
        <li><a href="#">分离的链接</a></li>
    </ul>
</div>
<div class="btn-group">
    <button type="button" class="btn btn-primary">原始</button>
    <button type="button" class="btn btn-primary dropdown-toggle"
        data-toggle="dropdown">
        <span class="caret"></span>
        <span class="sr-only">切换下拉菜单</span>
    </button>
    <ul class="dropdown-menu" role="menu">
        <li><a href="#">功能</a></li>
        <li><a href="#">另一个功能</a></li>
        <li><a href="#">其他</a></li>
        <li class="divider"></li>
        <li><a href="#">分离的链接</a></li>
    </ul>
</div>

```

</body>

</html>

结果如下所示:



按钮下拉菜单的大小

您可以使用带有各种大小按钮的下拉菜单: **.btn-large**、**.btn-sm** 或 **.btn-xs**。

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Bootstrap 实例 - 按钮下拉菜单的大小</title>
```

```
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```
    <script src="/scripts/jquery.min.js"></script>
```

```
    <script src="/bootstrap/js/bootstrap.min.js"></script>
```

```
</head>
```

```
<body>
```

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

```
    <button type="button" class="btn btn-default dropdown-toggle btn-lg"
        data-toggle="dropdown">
```

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

```
    </button>
```

```
    <ul class="dropdown-menu" role="menu">
```

```

        <li><a href="#">功能</a></li>
        <li><a href="#">另一个功能</a></li>
        <li><a href="#">其他</a></li>
        <li class="divider"></li>
        <li><a href="#">分离的链接</a></li>
    </ul>
</div>
<div class="btn-group">
    <button type="button" class="btn btn-primary dropdown-toggle btn-sm"
        data-toggle="dropdown">
        原始 <span class="caret"></span>
    </button>
    <ul class="dropdown-menu" role="menu">
        <li><a href="#">功能</a></li>
        <li><a href="#">另一个功能</a></li>
        <li><a href="#">其他</a></li>
        <li class="divider"></li>
        <li><a href="#">分离的链接</a></li>
    </ul>
</div>
<div class="btn-group">
    <button type="button" class="btn btn-success dropdown-toggle btn-xs"
        data-toggle="dropdown">
        成功 <span class="caret"></span>
    </button>
    <ul class="dropdown-menu" role="menu">
        <li><a href="#">功能</a></li>
        <li><a href="#">另一个功能</a></li>
        <li><a href="#">其他</a></li>
        <li class="divider"></li>
        <li><a href="#">分离的链接</a></li>
    </ul>
</div>
</body>
</html>

```

结果如下所示：



按钮上拉菜单

菜单也可以往上拉伸的，只需要简单地向父 `.btn-group` 容器添加 `.dropup` 即可。

```

<!DOCTYPE html>
<html>

```



```

<head>
  <title>Bootstrap 实例 - 按钮上拉菜单</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="row" style="margin-left:50px; margin-top:200px">
  <div class="btn-group dropdown">
    <button type="button" class="btn btn-default dropdown-toggle"
      data-toggle="dropdown">
      默认 <span class="caret"></span>
    </button>
    <ul class="dropdown-menu" role="menu">
      <li><a href="#">功能</a></li>
      <li><a href="#">另一个功能</a></li>
      <li><a href="#">其他</a></li>
      <li class="divider"></li>
      <li><a href="#">分离的链接</a></li>
    </ul>
  </div>
  <div class="btn-group dropdown">
    <button type="button" class="btn btn-primary dropdown-toggle"
      data-toggle="dropdown">
      原始 <span class="caret"></span>
    </button>
    <ul class="dropdown-menu" role="menu">
      <li><a href="#">功能</a></li>
      <li><a href="#">另一个功能</a></li>
      <li><a href="#">其他</a></li>
      <li class="divider"></li>
      <li><a href="#">分离的链接</a></li>
    </ul>
  </div>
</div>

</body>
</html>

```

结果如下所示:



Bootstrap 输入框组

本章将讲解 Bootstrap 支持的另一个特性，输入框组。输入框组扩展自 [表单控件](#)。使用输入框组，您可以很容易地向基于文本的输入框添加作为前缀和后缀的文本或按钮。

通过向输入域添加前缀和后缀的内容，您可以向用户输入添加公共的元素。例如，您可以添加美元符号，或者在 Twitter 用户名前添加 @，或者应用程序接口所需要的其他公共的元素。

向 `.form-control` 添加前缀或后缀元素的步骤如下：

- 把前缀或后缀元素放在一个带有 `class .input-group` 的 `<div>` 中。
- 接着，在相同的 `<div>` 内，在 `class` 为 `.input-group-addon` 的 `` 内放置额外的内容。
- 把该 `` 放置在 `<input>` 元素的前面或者后面。

“为了保持跨浏览器的兼容性，请避免使用 `<select>` 元素，因为它们在 WebKit 浏览器中不能完全渲染出效果。也不要直接向表单组应用输入框组的 `class`，输入框组是一个孤立的组件。

基本的输入框组

下面的实例演示了基本的输入框组：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 基本的输入框组</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div style="padding: 100px 100px 10px;">
  <form class="bs-example bs-example-form" role="form">
    <div class="input-group">
      <span class="input-group-addon">@</span>
      <input type="text" class="form-control" placeholder="twitterhandle">
    </div>
    <br>
```

```

<div class="input-group">
  <input type="text" class="form-control">
  <span class="input-group-addon">.00</span>
</div>
<br>
<div class="input-group">
  <span class="input-group-addon">$</span>
  <input type="text" class="form-control">
  <span class="input-group-addon">.00</span>
</div>
</form>
</div>

```

</body>

</html>

结果如下所示：



输入框组的大小

您可以通过向 **.input-group** 添加相对表单大小的 class（比如 **.input-group-lg**、**input-group-sm**、**input-group-xs**）来改变输入框组的大小。输入框中的内容会自动调整大小。

下面的实例演示了这点：

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Bootstrap 实例 - 输入框组的大小</title>
```

```
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```
  <script src="/scripts/jquery.min.js"></script>
```

```
  <script src="/bootstrap/js/bootstrap.min.js"></script>
```

```
</head>
```

```
<body>
```

```
<div style="padding: 100px 100px 10px;">
```

```
  <form class="bs-example bs-example-form" role="form">
```

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

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

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

```

</div><br>

<div class="input-group">
  <span class="input-group-addon">@</span>
  <input type="text" class="form-control" placeholder="Twitterhandle">
</div><br>

<div class="input-group input-group-sm">
  <span class="input-group-addon">@</span>
  <input type="text" class="form-control" placeholder="Twitterhandle">
</div>
</form>
</div>

</body>
</html>

```

结果如下所示：



复选框和单选插件

您可以把复选框和单选插件作为输入框组的前缀或者后缀元素，如下面的实例所示：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 输入框组的复选框和单选插件</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div style="padding: 100px 100px 10px;">
  <form class="bs-example bs-example-form" role="form">
    <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 --><br>
<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 -->
</form>
</div>

```

</body>

</html>

结果如下所示:



按钮插件

您也可以把按钮作为输入框组的前缀或者后缀元素，这个时候您就不是添加 `.input-group-addon` class，您需要使用 class `.input-group-btn` 来包裹按钮。这是必需的，因为默认的浏览器样式不会被重写。下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 输入框组的按钮插件</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div style="padding: 100px 100px 10px;">
    <form class="bs-example bs-example-form" role="form">

```

```

<div class="row">
  <div class="col-lg-6">
    <div class="input-group">
      <span class="input-group-btn">
        <button class="btn btn-default" type="button">
          Go!
        </button>
      </span>
      <input type="text" class="form-control">
    </div><!-- /input-group -->
  </div><!-- /.col-lg-6 --><br>
  <div class="col-lg-6">
    <div class="input-group">
      <input type="text" class="form-control">
      <span class="input-group-btn">
        <button class="btn btn-default" type="button">
          Go!
        </button>
      </span>
    </div><!-- /input-group -->
  </div><!-- /.col-lg-6 -->
</div><!-- /.row -->
</form>
</div>

```

</body>

</html>

结果如下所示：



带有下拉菜单的按钮

在输入框组中添加带有下拉菜单的按钮，只需要简单地在一个 `.input-group-btn` class 中包裹按钮和下拉菜单即可，如下面的实例所示：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 输入框组的下拉菜单按钮</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>

```

```

    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div style="padding: 100px 100px 10px;">
  <form class="bs-example bs-example-form" role="form">
    <div class="row">
      <div class="col-lg-6">
        <div class="input-group">
          <div class="input-group-btn">
            <button type="button" class="btn btn-default
              dropdown-toggle" data-toggle="dropdown">
              下拉菜单
            <span class="caret"></span>
          </button>
          <ul class="dropdown-menu">
            <li><a href="#">功能</a></li>
            <li><a href="#">另一个功能</a></li>
            <li><a href="#">其他</a></li>
            <li class="divider"></li>
            <li><a href="#">分离的链接</a></li>
          </ul>
        </div><!-- /btn-group -->
        <input type="text" class="form-control">
      </div><!-- /.col-lg-6 --><br>
      <div class="col-lg-6">
        <div class="input-group">
          <input type="text" class="form-control">
          <div class="input-group-btn">
            <button type="button" class="btn btn-default
              dropdown-toggle" data-toggle="dropdown">
              下拉菜单
            <span class="caret"></span>
          </button>
          <ul class="dropdown-menu pull-right">
            <li><a href="#">功能</a></li>
            <li><a href="#">另一个功能</a></li>
            <li><a href="#">其他</a></li>
            <li class="divider"></li>
            <li><a href="#">分离的链接</a></li>
          </ul>
        </div><!-- /btn-group -->
      </div><!-- /input-group -->
    </div>
  </form>
</div>

```

```

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

```

```

</body>
</html>

```

结果如下所示：



分割的下拉菜单按钮

在输入框组中添加带有下拉菜单的分割按钮，使用与下拉菜单按钮大致相同的样式，但是对下拉菜单添加了主要的功能，如下面的实例所示：

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 输入框组中分割的下拉菜单按钮</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div style="padding: 100px 100px 10px;">
    <form class="bs-example bs-example-form" role="form">
        <div class="row">
            <div class="col-lg-6">
                <div class="input-group">
                    <div class="input-group-btn">
                        <button type="button" class="btn btn-default"
                            tabindex="-1">下拉菜单
                        </button>
                        <button type="button" class="btn btn-default
                            dropdown-toggle" data-toggle="dropdown" tabindex="-1">
                            <span class="caret"></span>
                            <span class="sr-only">切换下拉菜单</span>
                        </button>
                        <ul class="dropdown-menu">
                            <li><a href="#">功能</a></li>
                            <li><a href="#">另一个功能</a></li>

```



```

        <li><a href="#">其他</a></li>
        <li class="divider"></li>
        <li><a href="#">分离的链接</a></li>
    </ul>
</div><!-- /btn-group -->
<input type="text" class="form-control">
</div><!-- /input-group -->
</div><!-- /.col-lg-6 --><br>
<div class="col-lg-6">
    <div class="input-group">
        <input type="text" class="form-control">
        <div class="input-group-btn">
            <button type="button" class="btn btn-default"
                tabindex="-1">下拉菜单
            </button>
            <button type="button" class="btn btn-default
                dropdown-toggle" data-toggle="dropdown" tabindex="-1">
                <span class="caret"></span>
                <span class="sr-only">切换下拉菜单</span>
            </button>
            <ul class="dropdown-menu pull-right">
                <li><a href="#">功能</a></li>
                <li><a href="#">另一个功能</a></li>
                <li><a href="#">其他</a></li>
                <li class="divider"></li>
                <li><a href="#">分离的链接</a></li>
            </ul>
        </div><!-- /btn-group -->
    </div><!-- /input-group -->
</div><!-- /.col-lg-6 -->
</div><!-- /.row -->
</form>
</div>

</body>
</html>

```

结果如下所示:



The image displays two examples of Bootstrap input groups. The first example shows a text input field followed by a dropdown button labeled '下拉菜单' (Dropdown Menu). The second example shows a text input field followed by a dropdown button labeled '下拉菜单' (Dropdown Menu).

Bootstrap 导航元素

本章我们将讲解 Bootstrap 提供的用于定义导航元素的一些选项。它们使用相同的标记和基类 `.nav`。Bootstrap 也提供了一个用于共享标记和状态的帮助器类。改变修饰的 `class`，可以在不同的样式间进行切换。

表格导航或标签

创建一个标签式的导航菜单：

- 以一个带有 `class .nav` 的无序列表开始。
- 添加 `class .nav-tabs`。

下面的实例演示了这点：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 标签式的导航菜单</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<p>标签式的导航菜单</p>
<ul class="nav nav-tabs">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">SVN</a></li>
  <li><a href="#">iOS</a></li>
  <li><a href="#">VB.Net</a></li>
  <li><a href="#">Java</a></li>
  <li><a href="#">PHP</a></li>
</ul>

</body>
</html>
```

结果如下所示：

标签式的导航菜单



胶囊式的导航菜单

基本的胶囊式导航菜单

如果需要把标签改成胶囊的样式，只需要使用 `class .nav-pills` 代替 `.nav-tabs` 即可，其他的步骤与上面相同。

下面的实例演示了这点：

```
<!DOCTYPE html>
<html>
<head>
```

```

<title>Bootstrap 实例 - 基本的胶囊式导航菜单</title>
<link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<script src="/scripts/jquery.min.js"></script>
<script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<p>基本的胶囊式导航菜单</p>
<ul class="nav nav-pills">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">SVN</a></li>
  <li><a href="#">iOS</a></li>
  <li><a href="#">VB.Net</a></li>
  <li><a href="#">Java</a></li>
  <li><a href="#">PHP</a></li>
</ul>

</body>
</html>

```

结果如下所示：

基本的胶囊式导航菜单



垂直的胶囊式导航菜单

您可以在使用 class **.nav**、**.nav-pills** 的同时使用 class **.nav-stacked**，让胶囊垂直堆叠。

下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 垂直的胶囊式导航菜单</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<p>垂直的胶囊式导航菜单</p>
<ul class="nav nav-pills nav-stacked">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">SVN</a></li>
  <li><a href="#">iOS</a></li>
  <li><a href="#">VB.Net</a></li>

```

```

    <li><a href="#">Java</a></li>
    <li><a href="#">PHP</a></li>
</ul>

```

```
</body>
```

```
</html>
```

结果如下所示：

垂直的胶囊式导航菜单



两端对齐的导航

您可以在屏幕宽度大于 768px 时，通过在分别使用 `.nav`、`.nav-tabs` 或 `.nav`、`.nav-pills` 的同时使用 class `.nav-justified`，让标签式或胶囊式导航菜单与父元素等宽。在更小的屏幕上，导航链接会堆叠。

下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 两端对齐的导航元素</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<p>两端对齐的导航元素</p>
<ul class="nav nav-pills nav-justified">
    <li class="active"><a href="#">Home</a></li>
    <li><a href="#">SVN</a></li>
    <li><a href="#">iOS</a></li>
    <li><a href="#">VB.Net</a></li>
    <li><a href="#">Java</a></li>
    <li><a href="#">PHP</a></li>

```

```
</ul><br><br><br>
```

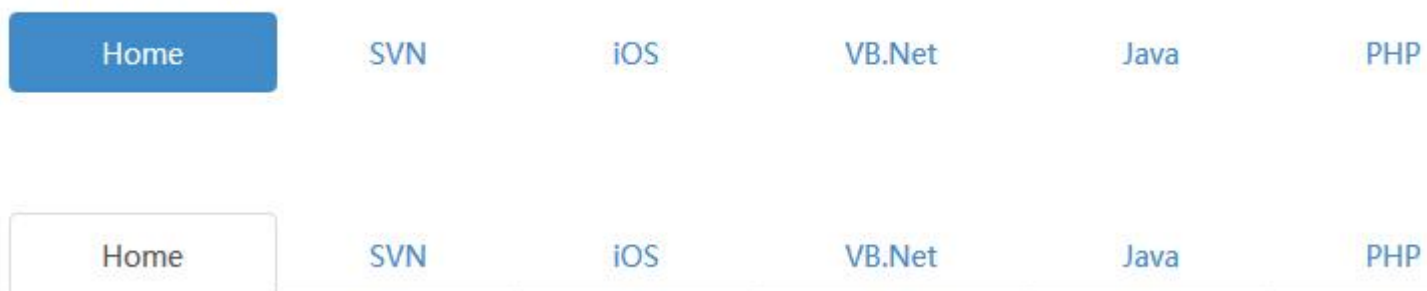
```
<ul class="nav nav-tabs nav-justified">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">SVN</a></li>
  <li><a href="#">iOS</a></li>
  <li><a href="#">VB.Net</a></li>
  <li><a href="#">Java</a></li>
  <li><a href="#">PHP</a></li>
</ul>
```

```
</body>
```

```
</html>
```

结果如下所示:

两端对齐的导航元素



禁用链接

对每个 **.nav** class，如果添加了 **.disabled** class，则会创建一个灰色的链接，同时禁用了该链接的 **:hover** 状态，如下面的实例所示：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 导航元素中的禁用链接</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>
```

<p>导航元素中的禁用链接</p>

```
<ul class="nav nav-pills">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">SVN</a></li>
  <li class="disabled"><a href="#">iOS（禁用链接）</a></li>
  <li><a href="#">VB.Net</a></li>
  <li><a href="#">Java</a></li>
```

```

    <li><a href="#">PHP</a></li>
</ul><br><br>

```

```

<ul class="nav nav-tabs">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">SVN</a></li>
  <li><a href="#">iOS</a></li>
  <li class="disabled"><a href="#">VB.Net（禁用链接）</a></li>
  <li><a href="#">Java</a></li>
  <li><a href="#">PHP</a></li>
</ul>

```

```
</body>
```

```
</html>
```

结果如下所示：

导航元素中的禁用链接



“该 class 只会改变 <a> 的外观，不会改变它的功能。在这里，您需要使用自定义的 JavaScript 来禁用链接。

下拉菜单

导航菜单与下拉菜单使用相似的语法。默认情况下，列表项的锚与一些数据属性协同合作来触发带有 **.dropdown-menu** class 的无序列表。

带有下拉菜单的标签

向标签添加下拉菜单的步骤如下：

- 以一个带有 class **.nav** 的无序列表开始。
- 添加 class **.nav-tabs**。
- 添加带有 **.dropdown-menu** class 的无序列表。

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 带有下拉菜单的标签</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

```

<p>带有下拉菜单的标签</p>

```
<ul class="nav nav-tabs">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">SVN</a></li>
  <li><a href="#">iOS</a></li>
  <li><a href="#">VB.Net</a></li>
  <li class="dropdown">
    <a class="dropdown-toggle" data-toggle="dropdown" href="#">
      Java <span class="caret"></span>
    </a>
    <ul class="dropdown-menu">
      <li><a href="#">Swing</a></li>
      <li><a href="#">jMeter</a></li>
      <li><a href="#">EJB</a></li>
      <li class="divider"></li>
      <li><a href="#">分离的链接</a></li>
    </ul>
  </li>
  <li><a href="#">PHP</a></li>
</ul>
```

</body>

</html>

结果如下所示:

带有下拉菜单的标签



带有下拉菜单的胶囊

步骤与创建带有下拉菜单的标签相同，只是需要把 **.nav-tabs** class 改为 **.nav-pills**，如下面的实例所示:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Bootstrap 实例 - 带有下拉菜单的胶囊</title>
```

```
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```
  <script src="/scripts/jquery.min.js"></script>
```

```

    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<p>带有下拉菜单的胶囊</p>
<ul class="nav nav-pills">
    <li class="active"><a href="#">Home</a></li>
    <li><a href="#">SVN</a></li>
    <li><a href="#">iOS</a></li>
    <li><a href="#">VB.Net</a></li>
    <li class="dropdown">
        <a class="dropdown-toggle" data-toggle="dropdown" href="#">
            Java <span class="caret"></span>
        </a>
        <ul class="dropdown-menu">
            <li><a href="#">Swing</a></li>
            <li><a href="#">jMeter</a></li>
            <li><a href="#">EJB</a></li>
            <li class="divider"></li>
            <li><a href="#">分离的链接</a></li>
        </ul>
    </li>
    <li><a href="#">PHP</a></li>
</ul>

</body>
</html>

```

结果如下所示：

带有下拉菜单的胶囊



Bootstrap 导航栏

导航栏是一个很好的功能，是 Bootstrap 网站的一个突出特点。导航栏在您的应用或网站中作为导航页头

的响应式基础组件。导航栏在移动设备的视图中是折叠的，随着可用视口宽度的增加，导航栏也会水平展开。在 Bootstrap 导航栏的核心中，导航栏包括了站点名称和基本的导航定义样式。

默认的导航栏

创建一个默认的导航栏的步骤如下：

- 向 `<nav>` 标签添加 class `.navbar`、`.navbar-default`。
- 向上面的元素添加 `role="navigation"`，有助于增加可访问性。
- 向 `<div>` 元素添加一个标题 class `.navbar-header`，内部包含了带有 class `navbar-brand` 的 `<a>` 元素。这会让文本看起来更大一号。
- 为了向导航栏添加链接，只需要简单地添加带有 class `.nav`、`.navbar-nav` 的无序列表即可。

下面的实例演示了这点：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 默认的导航栏</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<nav class="navbar navbar-default" role="navigation">
  <div class="navbar-header">
    <a class="navbar-brand" href="#">W3Cschool</a>
  </div>
  <div>
    <ul class="nav navbar-nav">
      <li class="active"><a href="#">iOS</a></li>
      <li><a href="#">SVN</a></li>
      <li class="dropdown">
        <a href="#" class="dropdown-toggle" data-toggle="dropdown">
          Java
          <b class="caret"></b>
        </a>
        <ul class="dropdown-menu">
          <li><a href="#">jmeter</a></li>
          <li><a href="#">EJB</a></li>
          <li><a href="#">Jasper Report</a></li>
          <li class="divider"></li>
          <li><a href="#">分离的链接</a></li>
          <li class="divider"></li>
          <li><a href="#">另一个分离的链接</a></li>
        </ul>
      </li>
    </ul>
  </div>
</nav>
```

```

    </div>
</nav>

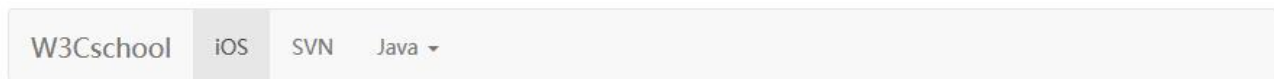
```

```

</body>
</html>

```

结果如下所示：



响应式的导航栏

为了给导航栏添加响应式特性，您要折叠的内容必须包裹在带有 classes **.collapse**、**.navbar-collapse** 的 `<div>` 中。折叠起来的导航栏实际上是一个带有 class **.navbar-toggle** 及两个 `data-` 元素的按钮。第一个是 **data-toggle**，用于告诉 JavaScript 需要对按钮做什么，第二个是 **data-target**，指示要切换到哪一个元素。三个带有 class **.icon-bar** 的 `` 创建所谓的汉堡按钮。这些会切换为 **.nav-collapse** `<div>` 中的元素。为了实现以上这些功能，您必须包含 [Bootstrap 折叠（Collapse）插件](#)。

下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 响应式的导航栏</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<nav class="navbar navbar-default" role="navigation">
  <div class="navbar-header">
    <button type="button" class="navbar-toggle" data-toggle="collapse"
      data-target="#example-navbar-collapse">
      <span class="sr-only">切换导航</span>
      <span class="icon-bar"></span>
      <span class="icon-bar"></span>
      <span class="icon-bar"></span>
    </button>
    <a class="navbar-brand" href="#">W3Cschool</a>
  </div>
  <div class="collapse navbar-collapse" id="example-navbar-collapse">
    <ul class="nav navbar-nav">
      <li class="active"><a href="#">iOS</a></li>
      <li><a href="#">SVN</a></li>
      <li class="dropdown">
        <a href="#" class="dropdown-toggle" data-toggle="dropdown">

```

```

        Java <b class="caret"></b>
    </a>
    <ul class="dropdown-menu">
        <li><a href="#">jmeter</a></li>
        <li><a href="#">EJB</a></li>
        <li><a href="#">Jasper Report</a></li>
        <li class="divider"></li>
        <li><a href="#">分离的链接</a></li>
        <li class="divider"></li>
        <li><a href="#">另一个分离的链接</a></li>
    </ul>
</li>
</ul>
</div>
</nav>

</body>
</html>

```

结果如下所示:



导航栏中的表单

导航栏中的表单不是使用 [Bootstrap 表单](#) 章节中所讲到的默认的 class，它是使用 **.navbar-form** class。这确保了表单适当的垂直对齐和在较窄的视口中折叠的行为。使用对齐方式选项（这将在组件对齐方式部分进行详细讲解）来决定导航栏中的内容放置在哪里。

下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 导航栏中的表单</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

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

```

```

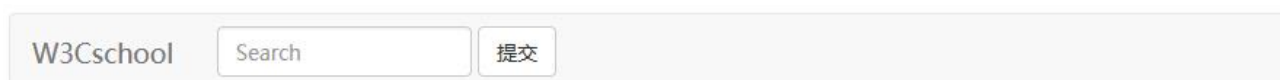
<div class="navbar-header">
  <a class="navbar-brand" href="#">W3Cschool</a>
</div>
<div>
  <form class="navbar-form navbar-left" role="search">
    <div class="form-group">
      <input type="text" class="form-control" placeholder="Search">
    </div>
    <button type="submit" class="btn btn-default">提交</button>
  </form>
</div>
</nav>

```

</body>

</html>

结果如下所示:



导航栏中的按钮

您可以使用 class **.navbar-btn** 向不在 `<form>` 中的 `<button>` 元素添加按钮，按钮在导航栏上垂直居中。**.navbar-btn** 可被使用在 `<a>` 和 `<input>` 元素上。

“不要在 **.navbar-nav** 内的 `<a>` 元素上使用 **.navbar-btn**，因为它不是标准的 [button class](#)。

下面的实例演示了这点:

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 导航栏中的按钮</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<nav class="navbar navbar-default" role="navigation">
  <div class="navbar-header">
    <a class="navbar-brand" href="#">W3Cschool</a>
  </div>
  <div>
    <form class="navbar-form navbar-left" role="search">
      <div class="form-group">
        <input type="text" class="form-control" placeholder="Search">
      </div>
    </form>
  </div>
</nav>

```

```

        </div>
        <button type="submit" class="btn btn-default">提交按钮</button>
    </form>
    <button type="button" class="btn btn-default navbar-btn">
        导航栏按钮
    </button>
</div>
</nav>

</body>
</html>

```

结果如下所示：

导航栏中的文本

如果需要在导航中包含文本字符串，请使用 `class .navbar-text`。这通常与 `<p>` 标签一起使用，确保适当的前导和颜色。下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 导航栏中的文本</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<nav class="navbar navbar-default" role="navigation">
    <div class="navbar-header">
        <a class="navbar-brand" href="#">W3Cschool</a>
    </div>
    <div>
        <p class="navbar-text">Signed in as Thomas</p>
    </div>
</nav>

</body>
</html>

```

结果如下所示：

非导航链接

如果您不想在常规的导航栏导航组件内使用标准的链接, 那么请使用 `class navbar-link` 来为默认的和倒转的导航栏选项添加适当的颜色, 如下面的实例所示:

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 非导航链接</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<nav class="navbar navbar-default" role="navigation">
  <div class="navbar-header">
    <a class="navbar-brand" href="#">W3Cschool</a>
  </div>
  <div>
    <p class="navbar-text navbar-right">Signed in as
      <a href="#" class="navbar-link">Thomas</a>
    </p>
  </div>
</div>

</body>
</html>
```

结果如下所示:



组件对齐方式

您可以使用实用工具 `class .navbar-left` 或 `.navbar-right` 向左或向右对齐导航栏中的 *导航链接、表单、按钮或文本* 这些组件。这两个 `class` 都会在指定的方向上添加 CSS 浮动。下面的实例演示了这点:

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 组件对齐方式</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<nav class="navbar navbar-default" role="navigation">
  <div class="navbar-header">
```

```

    <a class="navbar-brand" href="#">W3Cschool</a>
</div>
<div>
    <!--向左对齐-->
    <ul class="nav navbar-nav navbar-left">
        <li class="dropdown">
            <a href="#" class="dropdown-toggle" data-toggle="dropdown">
                Java
                <b class="caret"></b>
            </a>
            <ul class="dropdown-menu">
                <li><a href="#">jmeter</a></li>
                <li><a href="#">EJB</a></li>
                <li><a href="#">Jasper Report</a></li>
                <li class="divider"></li>
                <li><a href="#">分离的链接</a></li>
                <li class="divider"></li>
                <li><a href="#">另一个分离的链接</a></li>
            </ul>
        </li>
    </ul>
    <form class="navbar-form navbar-left" role="search">
        <button type="submit" class="btn btn-default">
            向左对齐-提交按钮
        </button>
    </form>
    <p class="navbar-text navbar-left">向左对齐-文本</p>
    <!--向右对齐-->
    <ul class="nav navbar-nav navbar-right">
        <li class="dropdown">
            <a href="#" class="dropdown-toggle" data-toggle="dropdown">
                Java <b class="caret"></b>
            </a>
            <ul class="dropdown-menu">
                <li><a href="#">jmeter</a></li>
                <li><a href="#">EJB</a></li>
                <li><a href="#">Jasper Report</a></li>
                <li class="divider"></li>
                <li><a href="#">分离的链接</a></li>
                <li class="divider"></li>
                <li><a href="#">另一个分离的链接</a></li>
            </ul>
        </li>
    </ul>

```

```

        <form class="navbar-form navbar-right" role="search">
            <button type="submit" class="btn btn-default">
                向右对齐-提交按钮
            </button>
        </form>
        <p class="navbar-text navbar-right">向右对齐-文本</p>
    </div>
</nav>

```

```

</body>
</html>

```

结果如下所示:



固定到顶部

Bootstrap 导航栏可以动态定位。默认情况下，它是块级元素，它是基于在 HTML 中放置的位置定位的。通过一些帮助器类，您可以把它放置在页面的顶部或者底部，或者您可以让它成为随着页面一起滚动的静态导航栏。

如果您想要让导航栏固定在页面的顶部，请向 **.navbar class** 添加 class **.navbar-fixed-top**。下面的实例演示了这点：

“为了防止导航栏与页面主体中的其他内容的顶部相交错，请向 **<body>** 标签添加至少 50 像素的内边距（padding），内边距的值可以根据您的需要进行设置。

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 固定到顶部</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<nav class="navbar navbar-default navbar-fixed-top" role="navigation">
    <div class="navbar-header">
        <a class="navbar-brand" href="#">W3Cschool</a>
    </div>
    <div>
        <ul class="nav navbar-nav">
            <li class="active"><a href="#">iOS</a></li>
            <li><a href="#">SVN</a></li>
            <li class="dropdown">
                <a href="#" class="dropdown-toggle" data-toggle="dropdown">

```

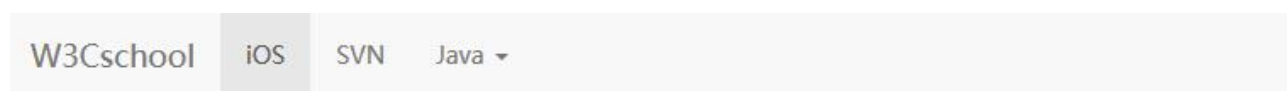


```

        Java <b class="caret"></b>
    </a>
    <ul class="dropdown-menu">
        <li><a href="#">jmeter</a></li>
        <li><a href="#">EJB</a></li>
        <li><a href="#">Jasper Report</a></li>
        <li class="divider"></li>
        <li><a href="#">分离的链接</a></li>
        <li class="divider"></li>
        <li><a href="#">另一个分离的链接</a></li>
    </ul>
</li>
</ul>
</div>
</nav>

</body>
</html>
结果如下所示:

```



固定到底部

如果您想要让导航栏固定在页面的底部，请向 **.navbar class** 添加 **class .navbar-fixed-bottom**。下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 固定到底部</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<nav class="navbar navbar-default navbar-fixed-bottom" role="navigation">
    <div class="navbar-header">
        <a class="navbar-brand" href="#">W3Cschool</a>
    </div>

```

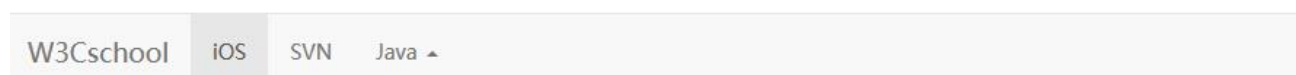
```

</div>
<div>
  <ul class="nav navbar-nav">
    <li class="active"><a href="#">iOS</a></li>
    <li><a href="#">SVN</a></li>
    <li class="dropdown">
      <a href="#" class="dropdown-toggle" data-toggle="dropdown">
        Java <b class="caret"></b>
      </a>
      <ul class="dropdown-menu">
        <li><a href="#">jmeter</a></li>
        <li><a href="#">EJB</a></li>
        <li><a href="#">Jasper Report</a></li>
        <li class="divider"></li>
        <li><a href="#">分离的链接</a></li>
        <li class="divider"></li>
        <li><a href="#">另一个分离的链接</a></li>
      </ul>
    </li>
  </ul>
</div>
</nav>

</body>
</html>

```

结果如下所示:



静态的顶部

如需创建能随着页面一起滚动的导航栏，请添加 **.navbar-static-top** class。该 class 不要求向 **<body>** 添加内边距（padding）。

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 静态的顶部</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>

```

```

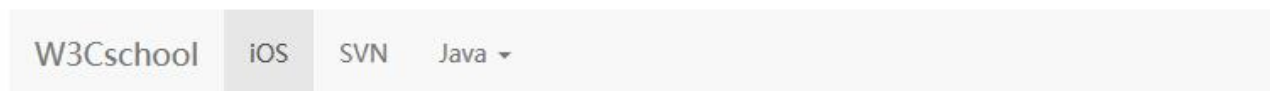
</head>
<body>

<nav class="navbar navbar-default navbar-static-top" role="navigation">
  <div class="navbar-header">
    <a class="navbar-brand" href="#">W3Cschool</a>
  </div>
  <div>
    <ul class="nav navbar-nav">
      <li class="active"><a href="#">iOS</a></li>
      <li><a href="#">SVN</a></li>
      <li class="dropdown">
        <a href="#" class="dropdown-toggle" data-toggle="dropdown">
          Java <b class="caret"></b>
        </a>
        <ul class="dropdown-menu">
          <li><a href="#">jmeter</a></li>
          <li><a href="#">EJB</a></li>
          <li><a href="#">Jasper Report</a></li>
          <li class="divider"></li>
          <li><a href="#">分离的链接</a></li>
          <li class="divider"></li>
          <li><a href="#">另一个分离的链接</a></li>
        </ul>
      </li>
    </ul>
  </div>
</nav>

</body>
</html>

```

结果如下所示：



倒置的导航栏

为了创建一个带有黑色背景白色文本的倒置的导航栏，只需要简单地向 `.navbar` class 添加 `.navbar-inverse` class 即可，如下面的实例所示：

“为了防止导航栏与页面主体中的其他内容的顶部相交错，请向 `<body>` 标签添加至少 50 像素的内边距（padding），内边距的值可以根据您的需要进行设置。

```

<!DOCTYPE html>
<html>
<head>

```

```

<title>Bootstrap 实例 - 倒置的导航栏</title>
<link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<script src="/scripts/jquery.min.js"></script>
<script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<nav class="navbar navbar-inverse" role="navigation">
  <div class="navbar-header">
    <a class="navbar-brand" href="#">W3Cschool</a>
  </div>
  <div>
    <ul class="nav navbar-nav">
      <li class="active"><a href="#">iOS</a></li>
      <li><a href="#">SVN</a></li>
      <li class="dropdown">
        <a href="#" class="dropdown-toggle" data-toggle="dropdown">
          Java <b class="caret"></b>
        </a>
        <ul class="dropdown-menu">
          <li><a href="#">jmeter</a></li>
          <li><a href="#">EJB</a></li>
          <li><a href="#">Jasper Report</a></li>
          <li class="divider"></li>
          <li><a href="#">分离的链接</a></li>
          <li class="divider"></li>
          <li><a href="#">另一个分离的链接</a></li>
        </ul>
      </li>
    </ul>
  </div>
</nav>

</body>
</html>

```

结果如下所示:



Bootstrap 面包屑导航（Breadcrumbs）

面包屑导航（Breadcrumbs）是一种基于网站层次信息的显示方式。以博客为例，面包屑导航可以显示发布日期、类别或标签。它们表示当前页面在导航层次结构内的位置。

Bootstrap 中的面包屑导航（Breadcrumbs）是一个简单的带有 `.breadcrumb` class 的无序列表。分隔符会通过 CSS（bootstrap.min.css）中下面所示的 class 自动被添加：

```
.breadcrumb > li + li:before {
    color: #CCCCCC;
    content: "/ ";
    padding: 0 5px;
}
```

下面的实例演示了面包屑导航：

```
<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 面包屑导航</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<ol class="breadcrumb">
    <li><a href="#">Home</a></li>
    <li><a href="#">2013</a></li>
    <li class="active">十一月</li>
</ol>

</body>
</html>
```

结果如下所示：



Bootstrap 分页

本章将讲解 Bootstrap 支持的分页特性。分页（Pagination），是一种无序列表，Bootstrap 像处理其他界面元素一样处理分页。

分页（Pagination）

下表列出了 Bootstrap 提供的处理分页的 class。

Class	描述	示例代码
<code>.pagination</code>	添加该 class 来在页面上显示分页。	<pre><ul class="pagination"> &laquo; 1 </pre>
<code>.disabled</code> , <code>.active</code>	您可以自定义链接，通过使用 <code>.disabled</code> 来定义不可点击的链	<pre><ul class="pagination"> <li class="disabled">&laquo;</pre>

	接，通过使用 .active 来指示当前的页面。	<pre><li class="active">1(current) </pre>
.pagination-lg, .pagination-sm	使用这些 class 来获取不同大小的项。	<pre><ul class="pagination pagination-lg">... <ul class="pagination">... <ul class="pagination pagination-sm">...</pre>

默认的分页

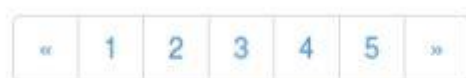
下面的实例演示了上表中所讨论的 class **.pagination** 的用法：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 默认的分页</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<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>

</body>
</html>
```

结果如下所示：



分页的状态

下面的实例演示了上表中所讨论的 class **.disabled**、**.active** 的用法：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 分页的状态</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
```

```

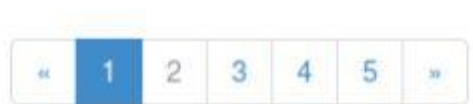
<body>

<ul class="pagination">
  <li><a href="#">&laquo;</a></li>
  <li class="active"><a href="#">1</a></li>
  <li class="disabled"><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>

</body>
</html>

```

结果如下所示：



分页的大小

下面的实例演示了上表中所讨论的 class **.pagination-*** 的用法：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 分页的大小</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<ul class="pagination pagination-lg">
  <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><br>
<ul class="pagination">
  <li><a href="#">&laquo;</a></li>
  <li><a href="#">1</a></li>

```

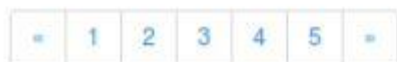
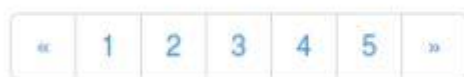
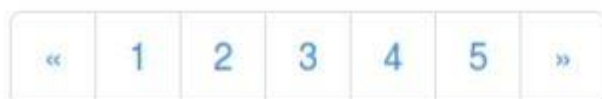
```

<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><br>
<ul class="pagination pagination-sm">
  <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>

</body>
</html>

```

结果如下所示：



翻页（Pager）

如果您想要创建一个简单的分页链接为用户提供导航，可通过翻页来实现。与分页链接一样，翻页也是无序列表。默认情况下，链接是居中显示。下表列出了 Bootstrap 处理翻页的 class。

Class	描述	示例代码
.pager	添加该 class 来获得翻页链接。	<pre> <ul class="pager"> Previous Next </pre>
.previous, .next	使用 class .previous 把链接向左对齐，使用 .next 把链接向右对齐。	<pre> <ul class="pager"> <li class="previous">&larr; Older <li class="next">Newer &rarr; </pre>
.disabled	添加该 class 来获得一个颜色变淡的外观。	<pre> <ul class="pager"> <li class="previous disabled">&larr; Older <li class="next">Newer &rarr; </pre>

默认的翻页

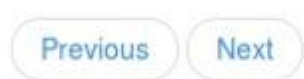
下面的实例演示了上表中所讨论的 class **.pager** 的用法：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 默认的翻页</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<ul class="pager">
  <li><a href="#">Previous</a></li>
  <li><a href="#">Next</a></li>
</ul>

</body>
</html>
```

结果如下所示：



对齐的链接

下面的实例演示了上表中所讨论的 class **.previous**、**.next** 的用法：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 翻页中对齐的链接</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<ul class="pager">
  <li class="previous"><a href="#">&larr; Older</a></li>
  <li class="next"><a href="#">Newer &rarr;</a></li>
</ul>

</body>
</html>
```

结果如下所示：

← Older

Newer →

翻页的状态

下面的实例演示了上表中所讨论的 class **.disabled** 的用法:

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 翻页的状态</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<ul class="pager">
  <li class="previous disabled"><a href="#">&larr; Older</a></li>
  <li class="next"><a href="#">Newer &rarr;</a></li>
</ul>

</body>
</html>
```

结果如下所示:

← Older

Newer →

Bootstrap 标签

本章将讲解 Bootstrap 标签。标签可用于计数、提示或页面上其他的标记显示。使用 class **.label** 来显示标签, 如下面的实例所示:

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 标签</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<h1>Example Heading <span class="label label-default">Label</span></h1>
```

```
<h2>Example Heading <span class="label label-default">Label</span></h2>
<h3>Example Heading <span class="label label-default">Label</span></h3>
<h4>Example Heading <span class="label label-default">Label</span></h4>
```

```
</body>
</html>
```

结果如下所示：



您可以使用修饰的 class **label-default**、**label-primary**、**label-success**、**label-info**、**label-warning**、**label-danger** 来改变标签的外观，如下面的实例所示：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 标签的变体</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
```

```
</head>
<body>
```

```
<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">危险标签</span>
```

```
</body>
</html>
```

结果如下所示：



Bootstrap 徽章 (Badges)

本章将讲解 Bootstrap 徽章 (Badges)。徽章与标签相似，主要的区别在于徽章的边角更加圆滑。

徽章 (Badges) 主要用于突出显示新的或未读的项。如需使用徽章，只需要把 `` 添加到链接、Bootstrap 导航等这些元素上即可。

下面的实例演示了这点：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 徽章 (Badges) </title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<a href="#">Mailbox <span class="badge">50</span></a>

</body>
</html>
```

结果如下所示：



当没有新的或未读的项时，通过 CSS 的 `:empty` 选择器，徽章会折叠起来，表示里边没有内容。

激活导航状态

您可以在激活状态的胶囊式导航和列表导航中放置徽章。通过使用 `` 来激活链接，如下面的实例所示：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 激活导航状态</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<h4>胶囊式导航中的激活状态</h4>
<ul class="nav nav-pills">
  <li class="active"><a href="#">首页 <span class="badge">42</span></a></li>
  <li><a href="#">简介</a></li>
  <li><a href="#">消息 <span class="badge">3</span></a></li>
</ul>
<br>
<h4>列表导航中的激活状态</h4>
<ul class="nav nav-pills nav-stacked" style="max-width: 260px;">
```

```

<li class="active">
  <a href="#">
    <span class="badge pull-right">42</span>
    首页
  </a>
</li>
<li><a href="#">简介</a></li>
<li>
  <a href="#">
    <span class="badge pull-right">3</span>
    消息
  </a>
</li>
</ul>

```

```

</body>
</html>

```

结果如下所示：

胶囊式导航中的激活状态



列表导航中的激活状态



Bootstrap 超大屏幕（Jumbotron）

本章将讲解 Bootstrap 支持的另一个特性，超大屏幕（Jumbotron）。顾名思义该组件可以增加标题的大小，并为登陆页面内容添加更多的外边距（margin）。使用超大屏幕（Jumbotron）的步骤如下：

- 创建一个带有 class **.jumbotron** 的容器 `<div>`。
- 除了更大的 `<h1>`，字体粗细 *font-weight* 被减为 200px。

下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 超大屏幕（Jumbotron）</title>

```

```

<link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<script src="/scripts/jquery.min.js"></script>
<script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="container">
  <div class="jumbotron">
    <h1>欢迎登陆页面！ </h1>
    <p>这是一个超大屏幕（Jumbotron）的实例。</p>
    <p><a class="btn btn-primary btn-lg" role="button">
      学习更多</a>
    </p>
  </div>
</div>

</body>
</html>

```

结果如下所示：



为了获得占用全部宽度且不帶圆角的超大屏幕，请在所有的 **.container** class 外使用 **.jumbotron** class，如下面的实例所示：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 超大屏幕（Jumbotron）</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

```

```

<div class="jumbotron">
  <div class="container">
    <h1>欢迎登陆页面！ </h1>
    <p>这是一个超大屏幕（Jumbotron）的实例。</p>
    <p><a class="btn btn-primary btn-lg" role="button">
      学习更多</a>
    </p>
  </div>
</div>

</body>
</html>
结果如下所示：

```



Bootstrap 页面标题（Page Header）

页面标题（Page Header）是个不错的功能，它会在网页标题四周添加适当的间距。当一个网页中有多个标题且每个标题之间需要添加一定的间距时，页面标题这个功能就显得特别有用。如需使用页面标题（Page Header），请把您的标题放置在带有 class **.page-header** 的 **<div>** 中：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 页面标题</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="page-header">
  <h1>页面标题实例
  <small>子标题</small>

```

```

    </h1>
</div>
<p>这是一个示例文本。这是一个示例文本。这是一个示例文本。这是一个示例文本。这是一个示例文本。
</p>

</body>
</html>

```

结果如下所示：

页面标题实例 子标题

这是一个示例文本。这是一个示例文本。这是一个示例文本。这是一个示例文本。这是一个示例文本。

Bootstrap 缩略图

本章将讲解 Bootstrap 缩略图。大多数站点都需要在网格中布局图像、视频、文本等。Bootstrap 通过缩略图为此提供了一种简便的方式。使用 Bootstrap 创建缩略图的步骤如下：

- 在图像周围添加带有 `class .thumbnail` 的 `<a>` 标签。
- 这会添加四个像素的内边距（padding）和一个灰色的边框。
- 当鼠标悬停在图像上时，会动画显示出图像的轮廓。

下面的实例演示了默认的缩略图：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 缩略图</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="row">
  <div class="col-sm-6 col-md-3">
    <a href="#" class="thumbnail">
      
    </a>
  </div>
  <div class="col-sm-6 col-md-3">
    <a href="#" class="thumbnail">

```



```

        
    </a>
</div>
<div class="col-sm-6 col-md-3">
    <a href="#" class="thumbnail">
        
    </a>
</div>
<div class="col-sm-6 col-md-3">
    <a href="#" class="thumbnail">
        
    </a>
</div>
</div>

</body>
</html>

```

结果如下所示:



添加自定义的内容

现在我们有了一个基本的缩略图，我们可以向缩略图添加各种 HTML 内容，比如标题、段落或按钮。具体步骤如下：

- 把带有 `class .thumbnail` 的 `<a>` 标签改为 `<div>`。
- 在该 `<div>` 内，您可以添加任何您想要添加的东西。由于这是一个 `<div>`，我们可以使用默认的基于 `span` 的命名规则来调整大小。
- 如果您想要给多个图像进行分组，请把它们放置在一个无序列表中，且每个列表项向左浮动。

下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 自定义缩略图</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

```

```

<div class="row">
  <div class="col-sm-6 col-md-3">
    <div class="thumbnail">
      
    </div>
    <div class="caption">
      <h3>缩略图标签</h3>
      <p>一些示例文本。一些示例文本。</p>
      <p>
        <a href="#" class="btn btn-primary" role="button">
          按钮
        </a>
        <a href="#" class="btn btn-default" role="button">
          按钮
        </a>
      </p>
    </div>
  </div>
  <div class="col-sm-6 col-md-3">
    <div class="thumbnail">
      
    </div>
    <div class="caption">
      <h3>缩略图标签</h3>
      <p>一些示例文本。一些示例文本。</p>
      <p>
        <a href="#" class="btn btn-primary" role="button">
          按钮
        </a>
        <a href="#" class="btn btn-default" role="button">
          按钮
        </a>
      </p>
    </div>
  </div>
  <div class="col-sm-6 col-md-3">
    <div class="thumbnail">
      
    </div>
    <div class="caption">

```

```

<h3>缩略图标签</h3>
<p>一些示例文本。一些示例文本。</p>
<p>
  <a href="#" class="btn btn-primary" role="button">
    按钮
  </a>
  <a href="#" class="btn btn-default" role="button">
    按钮
  </a>
</p>
</div>
</div>
<div class="col-sm-6 col-md-3">
  <div class="thumbnail">
    
    </div>
    <div class="caption">
      <h3>缩略图标签</h3>
      <p>一些示例文本。一些示例文本。</p>
      <p>
        <a href="#" class="btn btn-primary" role="button">
          按钮
        </a>
        <a href="#" class="btn btn-default" role="button">
          按钮
        </a>
      </p>
    </div>
  </div>
</div>
</div>
</body>
</html>

```

结果如下所示：



缩略图标签

一些示例文本。一些示例文本。



缩略图标签

一些示例文本。一些示例文本。



缩略图标签

一些示例文本。一些示例文本。



缩略图标签

一些示例文本。一些示例文本。



Bootstrap 警告 (Alerts)

本章将讲解警告 (Alerts) 以及 Bootstrap 所提供的用于警告的 class。警告 (Alerts) 向用户提供了一种定义消息样式的方式。它们为典型的用户操作提供了上下文信息反馈。

您可以为警告框添加一个可选的关闭按钮。为了创建一个内联的可取消的警告框，请使用 [警告 \(Alerts\) jQuery 插件](#)。

您可以通过创建一个 `<div>`，并向其添加一个 `.alert` class 和四个上下文 class（即 `.alert-success`、`.alert-info`、`.alert-warning`、`.alert-danger`）之一，来添加一个基本的警告框。下面的实例演示了这点：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 警告 (Alerts)</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="alert alert-success">成功！很好地完成了提交。</div>
<div class="alert alert-info">信息！请注意这个信息。</div>
<div class="alert alert-warning">警告！请不要提交。</div>
<div class="alert alert-danger">错误！请进行一些更改。</div>

</body>
</html>
```

结果如下所示：

成功！很好地完成了提交。

信息！请注意这个信息。

警告！请不要提交。

错误！请进行一些更改。

可取消的警告 (Dismissal Alerts)

创建一个可取消的警告（Dismissal Alert）步骤如下：

- 通过创建一个 `<div>`，并向其添加一个 `.alert` class 和四个上下文 class（即 `.alert-success`、`.alert-info`、`.alert-warning`、`.alert-danger`）之一，来添加一个基本的警告框。
- 同时向上面的 `<div>` class 添加可选的 `.alert-dismissible`。
- 添加一个关闭按钮。

下面的实例演示了这点：

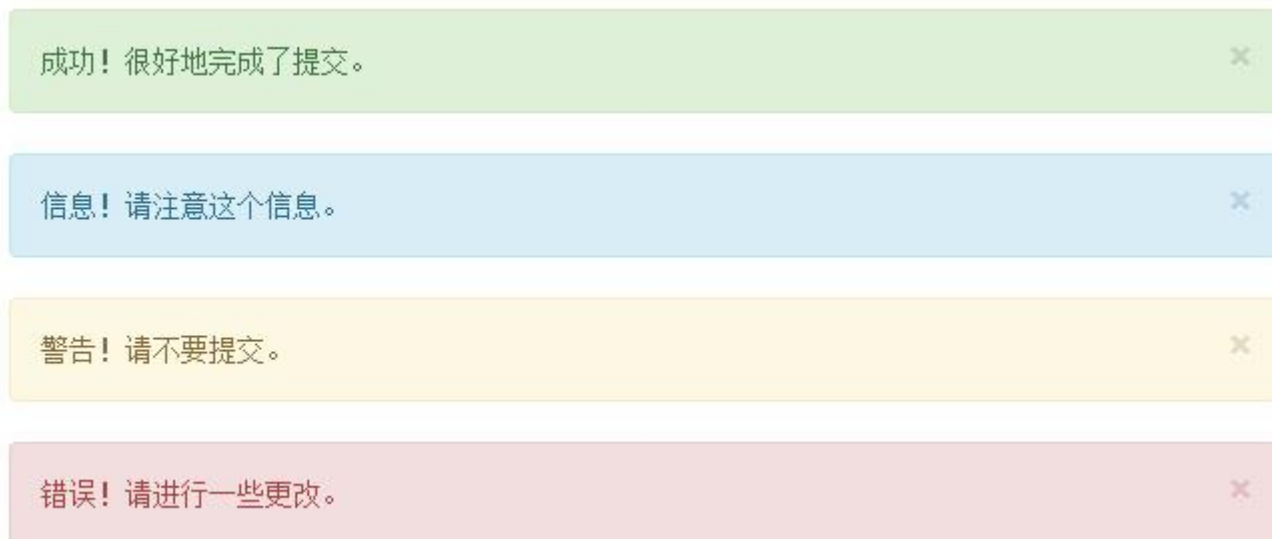
```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 可取消的警告（Dismissal Alerts）</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="alert alert-success alert-dismissible">
  <button type="button" class="close" data-dismiss="alert"
    aria-hidden="true">
    &times;
  </button>
  成功！很好地完成了提交。
</div>
<div class="alert alert-info alert-dismissible">
  <button type="button" class="close" data-dismiss="alert"
    aria-hidden="true">
    &times;
  </button>
  信息！请注意这个信息。
</div>
<div class="alert alert-warning alert-dismissible">
  <button type="button" class="close" data-dismiss="alert"
    aria-hidden="true">
    &times;
  </button>
  警告！请不要提交。
</div>
<div class="alert alert-danger alert-dismissible">
  <button type="button" class="close" data-dismiss="alert"
    aria-hidden="true">
    &times;
  </button>
  错误！请进行一些更改。
</div>
```

```
</body>
</html>
```

“请确保使用带有 `data-dismiss="alert"` data 属性的 `<button>` 元素。

结果如下所示：



警告（Alerts）中的链接

在警告（Alerts）中创建链接的步骤如下：

- 通过创建一个 `<div>`，并向其添加一个 `.alert` class 和四个上下文 class（即 `.alert-success`、`.alert-info`、`.alert-warning`、`.alert-danger`）之一，来添加一个基本的警告框。
- 使用 `.alert-link` 实体类来快速提供带有匹配颜色的链接。

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 警告（Alerts）中的链接</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="alert alert-success">
  <a href="#" class="alert-link">成功！很好地完成了提交。</a>
</div>
<div class="alert alert-info">
  <a href="#" class="alert-link">信息！请注意这个信息。</a>
</div>
<div class="alert alert-warning">
  <a href="#" class="alert-link">警告！请不要提交。</a>
```

```

</div>
<div class="alert alert-danger">
  <a href="#" class="alert-link">错误！ 请进行一些更改。</a>
</div>

```

```
</body>
```

```
</html>
```

结果如下所示：

成功！很好地完成了提交。

信息！请注意这个信息。

警告！请不要提交。

错误！请进行一些更改。

Bootstrap 进度条

本章将讲解 Bootstrap 进度条。在本教程中，您将看到如何使用 Bootstrap 创建加载、重定向或动作状态的进度条。

“Bootstrap 进度条使用 CSS3 过渡和动画来获得该效果。Internet Explorer 9 及之前的版本和旧版的 Firefox 不支持该特性，Opera 12 不支持动画。

默认的进度条

创建一个基本的进度条的步骤如下：

- 添加一个带有 class **.progress** 的 `<div>`。
- 接着，在上面的 `<div>` 内，添加一个带有 class **.progress-bar** 的空的 `<div>`。
- 添加一个带有百分比表示的宽度的 `style` 属性，例如 `style="60%"`；表示进度条在 60% 的位置。

让我们看看下面的实例：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 进度条</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>

```

```

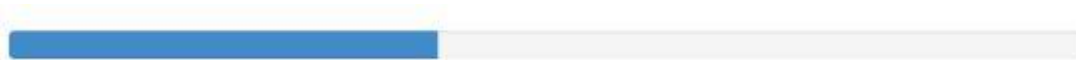
</head>
<body>

<div class="progress">
  <div class="progress-bar" role="progressbar" aria-valuenow="60"
    aria-valuemin="0" aria-valuemax="100" style="width: 40%;">
    <span class="sr-only">40% 完成</span>
  </div>
</div>

</body>
</html>

```

结果如下所示：



交替的进度条

创建不同样式的进度条的步骤如下：

- 添加一个带有 `class .progress` 的 `<div>`。
- 接着，在上面的 `<div>` 内，添加一个带有 `class .progress-bar` 和 `class progress-bar-*` 的空的 `<div>`。其中，* 可以是 **success**、**info**、**warning**、**danger**。
- 添加一个带有百分比表示的宽度的 `style` 属性，例如 `style="60%"`；表示进度条在 60% 的位置。

让我们看看下面的实例：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 交替的进度条</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="progress">
  <div class="progress-bar progress-bar-success" role="progressbar"
    aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
    style="width: 90%;">
    <span class="sr-only">90% 完成（成功）</span>
  </div>
</div>

<div class="progress">
  <div class="progress-bar progress-bar-info" role="progressbar"
    aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
    style="width: 30%;">
    <span class="sr-only">30% 完成（信息）</span>
  </div>
</div>

```

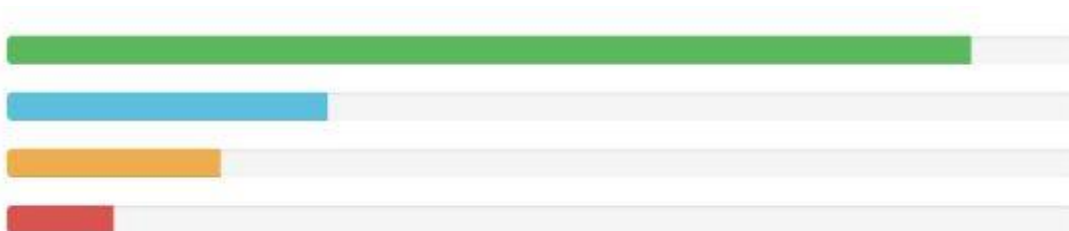


```

    </div>
</div>
<div class="progress">
  <div class="progress-bar progress-bar-warning" role="progressbar"
    aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
    style="width: 20%;">
    <span class="sr-only">20% 完成（警告）</span>
  </div>
</div>
<div class="progress">
  <div class="progress-bar progress-bar-danger" role="progressbar"
    aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
    style="width: 10%;">
    <span class="sr-only">10% 完成（危险）</span>
  </div>
</div>
</body>
</html>

```

结果如下所示：



条纹的进度条

创建一个条纹的进度条的步骤如下：

- 添加一个带有 class **.progress** 和 **.progress-striped** 的 **<div>**。
- 接着，在上面的 **<div>** 内，添加一个带有 class **.progress-bar** 和 class **progress-bar-*** 的空的 **<div>**。其中，* 可以是 **success**、**info**、**warning**、**danger**。
- 添加一个带有百分比表示的宽度的 **style** 属性，例如 **style="60%"**；表示进度条在 60% 的位置。

让我们看看下面的实例：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 条纹的进度条</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

```

```

<div class="progress progress-striped">
  <div class="progress-bar progress-bar-success" role="progressbar"
    aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
    style="width: 90%;">
    <span class="sr-only">90% 完成（成功） </span>
  </div>
</div>
<div class="progress progress-striped">
  <div class="progress-bar progress-bar-info" role="progressbar"
    aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
    style="width: 30%;">
    <span class="sr-only">30% 完成（信息） </span>
  </div>
</div>
<div class="progress progress-striped">
  <div class="progress-bar progress-bar-warning" role="progressbar"
    aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
    style="width: 20%;">
    <span class="sr-only">20% 完成（警告） </span>
  </div>
</div>
<div class="progress progress-striped">
  <div class="progress-bar progress-bar-danger" role="progressbar"
    aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
    style="width: 10%;">
    <span class="sr-only">10% 完成（危险） </span>
  </div>
</div>

</body>
</html>

```

结果如下所示：



动画的进度条

创建一个动画的进度条的步骤如下：

- 添加一个带有 class **.progress** 和 **.progress-striped** 的 `<div>`。同时添加 class **.active**。
- 接着，在上面的 `<div>` 内，添加一个带有 class **.progress-bar** 的空的 `<div>`。
- 添加一个带有百分比表示的宽度的 `style` 属性，例如 `style="60%"`；表示进度条在 60% 的位置。

这将会使条纹具有从右向左的运动感。

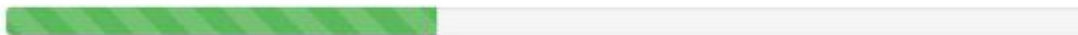
让我们看看下面的实例：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 动画的进度条</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="progress progress-striped active">
  <div class="progress-bar progress-bar-success" role="progressbar"
    aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
    style="width: 40%;">
    <span class="sr-only">40% 完成</span>
  </div>
</div>

</body>
</html>
```

结果如下所示：



堆叠的进度条

您甚至可以堆叠多个进度条。把多个进度条放在相同的 **.progress** 中即可实现堆叠，如下面的实例所示：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 堆叠的进度条</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="progress">
  <div class="progress-bar progress-bar-success" role="progressbar"
    aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
    style="width: 40%;">
    <span class="sr-only">40% 完成</span>
  </div>
```

```

<div class="progress-bar progress-bar-info" role="progressbar"
  aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
  style="width: 30%;">
  <span class="sr-only">30% 完成（信息）</span>
</div>
<div class="progress-bar progress-bar-warning" role="progressbar"
  aria-valuenow="60" aria-valuemin="0" aria-valuemax="100"
  style="width: 20%;">
  <span class="sr-only">20% 完成（警告）</span>
</div>
</div>

</body>
</html>

```

结果如下所示：



Bootstrap 多媒体对象（Media Object）

本章我们将讲解 Bootstrap 中的多媒体对象（Media Object）。这些抽象的对象样式用于创建各种类型的组件（比如：博客评论），我们可以在组件中使用图文混排，图像可以左对齐或者右对齐。媒体对象可以用更少的代码来实现媒体对象与文字的混排。

媒体对象轻量标记、易于扩展的特性是通过向简单的标记应用 class 来实现的。你可以在 HTML 标签中添加以下两种形式来设置媒体对象：

- **.media:** 该 class 允许将媒体对象里的多媒体（图像、视频、音频）浮动到内容区块的左边或者右边。
- **.media-list:** 如果你需要一个列表，各项内容是无序列表的一部分，可以使用该 class。可用于评论列表与文章列表。

让我们来看看下面这个有关默认的媒体对象 .media 的实例：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 默认的媒体对象</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="media">
  <a class="pull-left" href="#">
    
    </a>
    <div class="media-body">
        <h4 class="media-heading">媒体标题</h4>
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
    </div>
</div>
<div class="media">
    <a class="pull-left" href="#">
        
    </a>
    <div class="media-body">
        <h4 class="media-heading">媒体标题</h4>
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
    <div class="media">
        <a class="pull-left" href="#">
            
        </a>
        <div class="media-body">
            <h4 class="media-heading">媒体标题</h4>
            这是一些示例文本。这是一些示例文本。
            这是一些示例文本。这是一些示例文本。
            这是一些示例文本。这是一些示例文本。
            这是一些示例文本。这是一些示例文本。
            这是一些示例文本。这是一些示例文本。
        </div>
    </div>
</div>
</div>
</div>
</body>
</html>

```

结果如下所示：



媒体标题

这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。



媒体标题

这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。



媒体标题

这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。这是一些示例文本。

让我们来看看下面这个有关媒体对象列表 `.media-list` 的实例：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 媒体对象列表</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<ul class="media-list">
  <li class="media">
    <a class="pull-left" href="#">
      
    </a>
    <div class="media-body">
      <h4 class="media-heading">媒体标题</h4>
      <p>这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。</p>
      <!-- 嵌套的媒体对象 -->
      <div class="media">
        <a class="pull-left" href="#">
          
        </a>
        <div class="media-body">

```

```

<h4 class="media-heading">嵌套的媒体标题</h4>
这是一些示例文本。这是一些示例文本。
这是一些示例文本。这是一些示例文本。
这是一些示例文本。这是一些示例文本。
这是一些示例文本。这是一些示例文本。
这是一些示例文本。这是一些示例文本。
<!-- 嵌套的媒体对象 -->
<div class="media">
    <a class="pull-left" href="#">
        
    </a>
    <div class="media-body">
        <h4 class="media-heading">嵌套的媒体标题</h4>
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
    </div>
</div>
</div>
</div>
<!-- 嵌套的媒体对象 -->
<div class="media">
    <a class="pull-left" href="#">
        
    </a>
    <div class="media-body">
        <h4 class="media-heading">嵌套的媒体标题</h4>
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
        这是一些示例文本。这是一些示例文本。
    </div>
</div>
</div>
</li>
<li class="media">
    <a class="pull-right" href="#">
        

```

```

</a>
<div class="media-body">
  <h4 class="media-heading">媒体标题</h4>
  这是一些示例文本。这是一些示例文本。
  这是一些示例文本。这是一些示例文本。
  这是一些示例文本。这是一些示例文本。
  这是一些示例文本。这是一些示例文本。
  这是一些示例文本。这是一些示例文本。
</div>
</li>
</ul>

</body>
</html>

```

结果如下所示：



媒体标题

这是一些示例文本。这是一些示例文本。 这是一些示例文本。这是一些示例文本。 这是一些示例文本。
这是一些示例文本。 这是一些示例文本。 这是一些示例文本。 这是一些示例文本。 这是一些示例文本。



嵌套的媒体标题

这是一些示例文本。这是一些示例文本。 这是一些示例文本。这是一些示例文本。 这是一些示例文本。
这是一些示例文本。 这是一些示例文本。 这是一些示例文本。 这是一些示例文本。 这是一些示例文本。
这是一些示例文本。



嵌套的媒体标题

这是一些示例文本。这是一些示例文本。 这是一些示例文本。这是一些示例文本。 这是一些示例文本。
这是一些示例文本。 这是一些示例文本。 这是一些示例文本。 这是一些示例文本。 这是一些示例文本。
这是一些示例文本。 这是一些示例文本。



嵌套的媒体标题

这是一些示例文本。这是一些示例文本。 这是一些示例文本。这是一些示例文本。 这是一些示例文本。
这是一些示例文本。 这是一些示例文本。 这是一些示例文本。 这是一些示例文本。 这是一些示例文本。
这是一些示例文本。

媒体标题

这是一些示例文本。这是一些示例文本。 这是一些示例文本。这是一些示例文本。 这是一些示例文本。
这是一些示例文本。 这是一些示例文本。 这是一些示例文本。 这是一些示例文本。 这是一些示例文本。



Bootstrap 列表组

本章我们将讲解列表组。列表组件用于以列表形式呈现复杂的和自定义的内容。创建一个基本的列表组的步骤如下：

- 向元素 `` 添加 class `.list-group`。
- 向 `` 添加 class `.list-group-item`。

下面的实例演示了这点：

```
<!DOCTYPE html>
```



```

<html>
<head>
  <title>Bootstrap 实例 - 基本的列表组</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<ul class="list-group">
  <li class="list-group-item">免费域名注册</li>
  <li class="list-group-item">免费 Window 空间托管</li>
  <li class="list-group-item">图像的数量</li>
  <li class="list-group-item">24*7 支持</li>
  <li class="list-group-item">每年更新成本</li>
</ul>

</body>
</html>

```

结果如下所示：

免费域名注册
免费 Window 空间托管
图像的数量
24*7 支持
每年更新成本

向列表组添加徽章

我们可以向任意的列表项添加徽章组件，它会自动定位到右边。只需要在 `` 元素中添加 `` 即可。下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 向列表组添加徽章</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

```

```

<ul class="list-group">
  <li class="list-group-item">免费域名注册</li>
  <li class="list-group-item">免费 Window 空间托管</li>
  <li class="list-group-item">图像的数量</li>
  <li class="list-group-item">
    <span class="badge">新</span>
    24*7 支持
  </li>
  <li class="list-group-item">每年更新成本</li>
  <li class="list-group-item">
    <span class="badge">新</span>
    折扣优惠
  </li>
</ul>

```

```
</body>
```

```
</html>
```

结果如下所示：

免费域名注册	
免费 Window 空间托管	
图像的数量	
24*7 支持	新
每年更新成本	
折扣优惠	新

向列表组添加链接

通过使用锚标签代替列表项，我们可以向列表组添加链接。我们需要使用 `<div>` 代替 `` 元素。下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 向列表组添加链接</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>

```

```

<body>

<a href="#" class="list-group-item active">
    免费域名注册
</a>
<a href="#" class="list-group-item">24*7 支持</a>
<a href="#" class="list-group-item">免费 Window 空间托管</a>
<a href="#" class="list-group-item">图像的数量</a>
<a href="#" class="list-group-item">每年更新成本</a>

</body>
</html>

```

结果如下所示：

免费域名注册
24*7 支持
免费 Window 空间托管
图像的数量
每年更新成本

向列表组添加自定义内容

我们可以向上面已添加链接的列表组添加任意的 HTML 内容。下面的实例演示了这点：

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 向列表组添加自定义内容</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="list-group">
    <a href="#" class="list-group-item active">
        <h4 class="list-group-item-heading">
            入门网站包
        </h4>
    </a>
    <a href="#" class="list-group-item">
        <h4 class="list-group-item-heading">
            免费域名注册
        </h4>
        <p class="list-group-item-text">

```

您将通过网页进行免费域名注册。

</p>

<h4 class="list-group-item-heading">

24*7 支持

</h4>

<p class="list-group-item-text">

我们提供 24*7 支持。

</p>

</div>

<div class="list-group">

<h4 class="list-group-item-heading">

商务网站包

</h4>

<h4 class="list-group-item-heading">

免费域名注册

</h4>

<p class="list-group-item-text">

您将通过网页进行免费域名注册。

</p>

<h4 class="list-group-item-heading">24*7 支持</h4>

<p class="list-group-item-text">我们提供 24*7 支持。</p>

</div>

</body>

</html>

结果如下所示：

入门网站包
免费域名注册 您将通过网页进行免费域名注册。
24*7 支持 我们提供 24*7 支持。

商务网站包
免费域名注册 您将通过网页进行免费域名注册。
24*7 支持 我们提供 24*7 支持。

Bootstrap 面板（Panels）

本章将讲解 Bootstrap 面板（Panels）。面板组件用于把 DOM 组件插入到一个盒子中。创建一个基本的面板，只需要向 `<div>` 元素添加 `class .panel` 和 `class .panel-default` 即可，如下面的实例所示：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 默认的面板</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="panel panel-default">
  <div class="panel-body">
    这是一个基本的面板
  </div>
</div>

</body>
</html>
```

结果如下所示：

这是一个基本的面板

面板标题

我们可以通过以下两种方式来添加面板标题：

- 使用 `.panel-heading` class 可以很简单地向面板添加标题容器。to easily add a heading container to your panel.
- 使用带有 `.panel-title` class 的 `<h1>-<h6>` 来添加预定义样式的标题。

下面的实例演示了这两种方式：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 面板标题</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="panel panel-default">
  <div class="panel-heading">
    不帶 title 的面板标题
  </div>
  <div class="panel-body">
    面板内容
  </div>
</div>

<div class="panel panel-default">
  <div class="panel-heading">
    <h3 class="panel-title">
      帶有 title 的面板标题
    </h3>
  </div>
  <div class="panel-body">
    面板内容
  </div>
</div>

</body>
</html>
```

结果如下所示：



面板脚注

我们可以在面板中添加脚注，只需要把按钮或者副文本放在带有 class **.panel-footer** 的 `<div>` 中即可。下面的实例演示了这点：

```

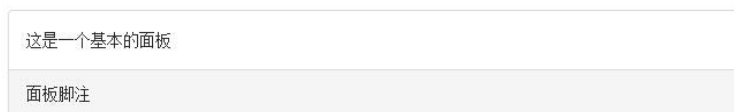
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 面板脚注</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="panel panel-default">
  <div class="panel-body">
    这是一个基本的面板
  </div>
  <div class="panel-footer">面板脚注</div>
</div>

</body>
</html>

```

结果如下所示：



“面板脚注不会从带语境色彩的面板中继承颜色和边框，因为它不是前景中的内容。”

带语境色彩的面板

使用语境状态类 **panel-primary**、**panel-success**、**panel-info**、**panel-warning**、**panel-danger**，来设置带语境色彩的面板，实例如下：

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 带语境色彩的面板</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="panel panel-primary">
  <div class="panel-heading">
    <h3 class="panel-title">面板标题</h3>
  </div>
  <div class="panel-body">
    这是一个基本的面板
  </div>
</div>

```

```
</div>
</div>
<div class="panel panel-success">
  <div class="panel-heading">
    <h3 class="panel-title">面板标题</h3>
  </div>
  <div class="panel-body">
    这是一个基本的面板
  </div>
</div>
<div class="panel panel-info">
  <div class="panel-heading">
    <h3 class="panel-title">面板标题</h3>
  </div>
  <div class="panel-body">
    这是一个基本的面板
  </div>
</div>
<div class="panel panel-warning">
  <div class="panel-heading">
    <h3 class="panel-title">面板标题</h3>
  </div>
  <div class="panel-body">
    这是一个基本的面板
  </div>
</div>
<div class="panel panel-danger">
  <div class="panel-heading">
    <h3 class="panel-title">面板标题</h3>
  </div>
  <div class="panel-body">
    这是一个基本的面板
  </div>
</div>

</body>
</html>
```

结果如下所示：



带表格的面板

为了在面板中创建一个无边框的表格，我们可以在面板中使用 `class .table`。假设有个 `<div>` 包含 `.panel-body`，我们可以向表格的顶部添加额外的边框用来分隔。如果没有包含 `.panel-body` 的 `<div>`，则组件会无中断地从面板头部移动到表格。

下面的实例演示了这点：

```
<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 带表格的面板</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="panel panel-default">
  <div class="panel-heading">
    <h3 class="panel-title">面板标题</h3>
  </div>
  <div class="panel-body">
    这是一个基本的面板
  </div>
  <table class="table">
    <th>产品</th><th>价格 </th>
    <tr><td>产品 A</td><td>200</td></tr>
    <tr><td>产品 B</td><td>400</td></tr>
```

```

    </table>
</div>
<div class="panel panel-default">
  <div class="panel-heading">面板标题</div>
  <table class="table">
    <th>产品</th><th>价格 </th>
    <tr><td>产品 A</td><td>200</td></tr>
    <tr><td>产品 B</td><td>400</td></tr>
  </table>
</div>

</body>
</html>

```

结果如下所示：

面板标题	
这是一个基本的面板	
产品	价格
产品 A	200
产品 B	400

面板标题	
产品	价格
产品 A	200
产品 B	400

带列表组的面板

我们可以在任何面板中包含列表组，通过在 `<div>` 元素中添加 `.panel` 和 `.panel-default` 类来创建面板，并在面板中添加列表组。您可以从 [列表组](#) 一章中学习如何创建列表组。

```

<!DOCTYPE html>
<html>
<head>
  <title>Bootstrap 实例 - 带列表组的面板</title>
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <script src="/scripts/jquery.min.js"></script>
  <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

<div class="panel panel-default">
  <div class="panel-heading">面板标题</div>
  <div class="panel-body">
    <p>这是一个基本的面板内容。这是一个基本的面板内容。

```

```

        这是一个基本的面板内容。这是一个基本的面板内容。
        这是一个基本的面板内容。这是一个基本的面板内容。
        这是一个基本的面板内容。这是一个基本的面板内容。
    </p>
</div>
<ul class="list-group">
    <li class="list-group-item">免费域名注册</li>
    <li class="list-group-item">免费 Window 空间托管</li>
    <li class="list-group-item">图像的数量</li>
    <li class="list-group-item">24*7 支持</li>
    <li class="list-group-item">每年更新成本</li>
</ul>
</div>
</body>
</html>

```

结果如下所示：

面板标题
这是一个基本的面板内容。这是一个基本的面板内容。 这是一个基本的面板内容。这是一个基本的面板内容。 这是一个基本的面板内容。这是一个基本的面板内容。 这是一个基本的面板内容。 这是一个基本的面板内容。
免费域名注册
免费 Window 空间托管
图像的数量
24*7 支持
每年更新成本

Bootstrap Well

Well 是一种会引起内容凹陷显示或插图效果的容器 `<div>`。为了创建 Well，只需要简单地把内容放在带有 class `.well` 的 `<div>` 中即可。下面的实例演示了一个默认的 Well：

```

<!DOCTYPE html>
<html>
<head>
    <title>Bootstrap 实例 - 默认的 Well</title>
    <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <script src="/scripts/jquery.min.js"></script>
    <script src="/bootstrap/js/bootstrap.min.js"></script>
</head>
<body>

    <div class="well">您好，我在 Well 中！ </div>

```

```
</body>
```

```
</html>
```

结果如下所示：



您好，我在 Well 中！

尺寸大小

您可以使用可选类 **well-lg** 或 **well-sm** 来改变 Well 的尺寸大小。这两个类是与 **.well** 类结合使用的。这两个类会影响内边距（padding），根据使用的类，Well 会显示得更大或者更小。

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Bootstrap 实例 - Well 的尺寸大小</title>
```

```
  <link href="/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```
  <script src="/scripts/jquery.min.js"></script>
```

```
  <script src="/bootstrap/js/bootstrap.min.js"></script>
```

```
</head>
```

```
<body>
```

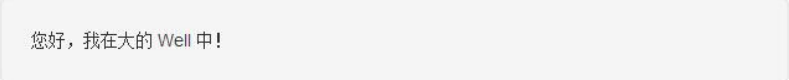
```
<div class="well well-lg">您好，我在大的 Well 中！ </div>
```

```
<div class="well well-sm">您好，我在小的 Well 中！ </div>
```

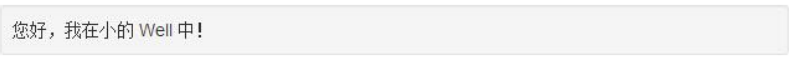
```
</body>
```

```
</html>
```

结果如下所示：



您好，我在大的 Well 中！



您好，我在小的 Well 中！