## Bootstrap 包的内容

* **基本结构**：Bootstrap 提供了一个带有网格系统、链接样式、背景的基本结构。这将在 **Bootstrap 基本结构** 部分详细讲解。
* **CSS**：Bootstrap 自带以下特性：全局的 CSS 设置、定义基本的 HTML 元素样式、可扩展的 class，以及一个先进的网格系统。这将在 **Bootstrap CSS** 部分详细讲解。
* **组件**：Bootstrap 包含了十几个可重用的组件，用于创建图像、下拉菜单、导航、警告框、弹出框等等。这将在 **布局组件** 部分详细讲解。
* **JavaScript 插件**：Bootstrap 包含了十几个自定义的 jQuery 插件。您可以直接包含所有的插件，也可以逐个包含这些插件。这将在 **Bootstrap 插件** 部分详细讲解。
* **定制**：您可以定制 Bootstrap 的组件、LESS 变量和 jQuery 插件来得到您自己的版本。

# Bootstrap 网格系统

网格系统（Grid System）。

Bootstrap 提供了一套响应式、移动设备优先的流式网格系统，随着屏幕或视口（viewport）尺寸的增加，系统会自动分为最多12列。

### 移动设备优先策略

* **内容**
  + 决定什么是最重要的。
* **布局**
  + 优先设计更小的宽度。
  + 基础的 CSS 是移动设备优先，媒体查询是针对于平板电脑、台式电脑。
* **渐进增强**
  + 随着屏幕大小的增加而添加元素。

响应式网格系统随着屏幕或视口（viewport）尺寸的增加，系统会自动分为最多12列。

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 4 | | | | 4 | | | | 4 | | | |
| 4 | | | | 8 | | | | | | | |
| 6 | | | | | | 6 | | | | | |
| 12 | | | | | | | | | | | |

## Bootstrap 网格系统（Grid System）的工作原理

网格系统通过一系列包含内容的行和列来创建页面布局。下面列出了 Bootstrap 网格系统是如何工作的：

* 行必须放置在 **.container** class 内，以便获得适当的对齐（alignment）和内边距（padding）。
* 内容应该放置在列内，且唯有列可以是行的直接子元素。
* 列通过内边距（padding）来创建列内容之间的间隙。该内边距是通过 **.rows** 上的外边距（margin）取负，表示第一列和最后一列的行偏移。
* 网格系统是通过指定您想要横跨的十二个可用的列来创建的。例如，要创建三个相等的列，则使用三个 **.col-xs-4**。

## 网格选项

下表总结了 Bootstrap 网格系统如何跨多个设备工作：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **超小设备手机（<768px）** | **小型设备平板电脑（≥768px）** | **中型设备台式电脑（≥992px）** | **大型设备台式电脑（≥1200px）** |
| 网格行为 | 一直是水平的 | 以折叠开始，断点以上是水平的 | 以折叠开始，断点以上是水平的 | 以折叠开始，断点以上是水平的 |
| 最大容器宽度 | None (auto) | 750px | 970px | 1170px |
| Class 前缀 | **.col-xs-** | **.col-sm-** | **.col-md-** | **.col-lg-** |
| 列数量和 | 12 | 12 | 12 | 12 |
| 最大列宽 | Auto | 60px | 78px | 95px |
| 间隙宽度 | 30px （一个列的每边分别 15px） | 30px （一个列的每边分别 15px） | 30px （一个列的每边分别 15px） | 30px （一个列的每边分别 15px） |
| 可嵌套 | Yes | Yes | Yes | Yes |
| 偏移量 | Yes | Yes | Yes | Yes |
| 列排序 | Yes | Yes | Yes | Yes |

### 基本的网格结构

下面是 Bootstrap 网格的基本结构：

<div class="container">

<div class="row">

<div class="col-\*-\*"></div>

<div class="col-\*-\*"></div>

</div>

<div class="row">...</div></div><div class="container">....

让我们来看几个简单的网格实例：

## 偏移列

<div class="row">

<div class="col-md-6 **col-md-offset-3**">..</div>

</div>

**.****col-md-offset-3** class 来居中这个 div。

## 嵌套列

为了在内容中嵌套默认的网格，请添加一个新的 **.row**，并在一个已有的 **.col-md-\*** 列内添加一组 **.col-md-\*** 列。被嵌套的行应包含一组列，这组列个数不能超过12（其实，没有要求你必须占满12列）。

在下面的实例中，布局有两个列，第二列被分为两行四个盒子。

<div class="container">

<div class="row">

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

<div class=“row”>

<div class="col-md-6 "></div>

<div class="col-md-6 "></div>

</div></div></div></div>

## 列排序

Bootstrap 网格系统另一个完美的特性，就是您可以很容易地以一种顺序编写列，然后以另一种顺序显示列。

您可以很轻易地改变带有 **.col-md-push-\*** 和 **.col-md-pull-\*** 类的内置网格列的顺序，其中 **\*** 范围是从 **1** 到 **11**。

在下面的实例中，我们有两列布局，左列很窄，作为侧边栏。我们将使用 **.col-md-push-\*** 和 **.col-md-pull-\*** 类来互换这两列的顺序。

# Bootstrap 排版

### 内联子标题

添加一个内联子标题,需要在元素两旁添加 <small>，或者添加 **.small** class，得到一个字号更小的颜色更浅的文本

<h1><small></small></h1>

## 强调

<small>本行内容是在标签内</small><br>

<strong>本行内容是在标签内</strong><br>

<em>本行内容是在标签内，并呈现为斜体</em><br>

<p class="text-left">向左对齐文本</p>

<p class="text-center">居中对齐文本</p>

<p class="text-right">向右对齐文本</p>

<p class="text-muted">本行内容是减弱的</p>

<p class="text-primary">本行内容带有一个 primary class</p>

<p class="text-success">本行内容带有一个 success class</p>

<p class="text-info">本行内容带有一个 info class</p>

<p class="text-warning">本行内容带有一个 warning class</p>

<p class="text-danger">本行内容带有一个 danger class</p>

## 缩写

Bootstrap 定义 <abbr> 元素的样式为显示在文本底部的一条虚线边框，当鼠标悬停在上面时会显示完整的文本，添加 .initialism 到 <abbr>得到一个更小字体的文本.

<abbr title="World Wide Web">WWW</abbr><br>

<abbr title="Real Simple Syndication" class="initialism">RSS</abbr>

## 地址（Address）

使用 <address> 标签，您可以在网页上显示联系信息。由于 <address> 默认为 display:block;，您需要使用 <br> 标签来为封闭的地址文本添加换行。

<address>

<strong>Full Name</strong><br>

<a href="mailto:#">mailto@somedomain.com</a>

</address>

# Bootstrap 代码

Bootstrap 允许您以两种方式显示代码：

* 第一种是 <code> 标签。如果您想要内联显示代码，那么您应该使用 <code> 标签。
* 第二种是 <pre> 标签。如果代码需要被显示为一个独立的块元素或者代码有多行，那么您应该使用 <pre> 标签。

请确保当您使用 <pre> 和 <code> 标签时，开始和结束标签使用了 unicode 变体： **&lt;** 和 **&gt;**。

|  |  |  |
| --- | --- | --- |
| **元素/类** | **描述** |  |
| <var> | 变量赋值: x = ab + y |  |
| <kbd> | 按键提示： CTRL + P |  |
| <pre> | 多行代码 |  |
| <pre class="pre-scrollable"> | 多行代码带有滚动条 |  |
| <samp> | 电脑程序输出: Sample output |  |
| <code> | 同一行代码片段: span, div |  |

# Bootstrap 表格

Bootstrap 提供了一个清晰的创建表格的布局。下表列出了 Bootstrap 支持的一些表格元素：

|  |  |
| --- | --- |
| **标签** | **描述** |
| <table> | 为表格添加基础样式。 |
| <thead> | 表格标题行的容器元素（<tr>），用来标识表格列。 |
| <tbody> | 表格主体中的表格行的容器元素（<tr>）。 |
| <tr> | 一组出现在单行上的表格单元格的容器元素（<td> 或 <th>）。 |
| <td> | 默认的表格单元格。 |
| <th> | 特殊的表格单元格，用来标识列或行（取决于范围和位置）。必须在 <thead> 内使用。 |
| <caption> | 关于表格存储内容的描述或总结。 |

### 表格类

下表样式可用于表格中：

|  |  |  |  |
| --- | --- | --- | --- |
| **类** | **描述** |  | |
| .table | 为任意 <table> 添加基本样式 (只有横向分隔线) |  | |
| .table-striped | 在 <tbody> 内添加斑马线形式的条纹 ( IE8 不支持) |  | |
| .table-bordered | 为所有表格的单元格添加边框 |  | |
| .table-hover | 在 <tbody> 内的任一行启用鼠标悬停状态 |  | |
| .table-condensed | 让表格更加紧凑 |  | |
| 联合使用所有表格类 | | |  |

### <tr>, <th> 和 <td> 类

下表的类可用于表格的行或者单元格：

|  |  |  |
| --- | --- | --- |
| **类** | **描述** |  |
| .active | 将悬停的颜色应用在行或者单元格上 |  |
| .success | 表示成功的操作 |  |
| .info | 表示信息变化的操作 |  |
| .warning | 表示一个警告的操作 |  |
| .danger | 表示一个危险的操作 |  |

## 基本的表格

要一个只带有内边距（padding）和水平分割的基本表，添加 class *.table*，

<table class="table">...</table>

## 可选的表格类

### 条纹表格

添加 *.table-striped* class，在 <tbody> 内的行上看到条纹，

<table class="table table-striped">

### 边框表格

添加 *.table-bordered* class，每个元素周围都有边框，且占整个表格是圆角，

<table class="table table-bordered">

### 悬停表格

添加 *.table-hover* class，当指针悬停在行上时会出现浅灰色背景

<table class="table table-hover">

### 精简表格

添加 .table-condensed class，行内边距（padding）切为两半，让表更紧凑

<table class="table table-condensed">

## 上下文类

下表中所列出的上下文类允许您改变表格行或单个单元格的背景颜色。

|  |  |
| --- | --- |
| **类** | **描述** |
| .active | 对某一特定的行或单元格应用悬停颜色 |
| .success | 表示一个成功的或积极的动作 |
| .warning | 表示一个需要注意的警告 |
| .danger | 表示一个危险的或潜在的负面动作 |

这些类可被应用到 <tr>、<td> 或 <th>。

## 响应式表格

通过把任意的 *.table* 包在 *.table-responsive* class 内，您可以让表格水平滚动以适应小型设备（小于 768px）

<div class="table-responsive">

<table class="table">

...

</table>

</div>

# Bootstrap 表单

## 表单布局

Bootstrap 提供了下列类型的表单布局：

* 垂直表单（默认）
* 内联表单
* 水平表单

### 垂直或基本表单

基本的表单结构是 Bootstrap 自带的，个别的表单控件自动接收一些全局样式。下面列出了创建基本表单的步骤：

* 向父 <form> 元素添加 *role="form"*。
* 把标签和控件放在一个带有 class *.form-group* 的 <div> 中。这是获取最佳间距所必需的。
* 向所有的文本元素 <input>、<textarea> 和 <select> 添加 class ="*form-control*" 。

<form role="**form**">

<div class="**form-group**">

<input type="text" class="**form-control**" id="name" placeholder="请输入名称">

</div>

</form>

### 内联表单

要创建一个表单，它的所有元素是内联的，向左对齐的，标签是并排的，向 <form> 标签添加 class *.form-inline*。

<form class="form-inline" role="form">

<div class="form-group"></div>

</form>

* 默认情况下，Bootstrap 中的 input、select 和 textarea 有 100% 宽度。在使用内联表单时，您需要在表单控件上设置一个宽度。
* 使用 class *.sr-only*，您可以隐藏内联表单的标签。

### 水平表单

水平表单与其他表单不仅标记的数量上不同，而且表单的呈现形式也不同。如需创建一个水平布局的表单，请按下面的几个步骤进行：

* 向父 <form> 元素添加 class *.form-horizontal*。
* 把标签和控件放在一个带有 class *.form-group* 的 <div> 中。
* 向标签添加 class *.control-label*。

<form class="form-horizontal" role="form">

<div class="form-group">

</div>

</form>

## 支持的表单控件

常见的表单控件*input、textarea、checkbox、radio 和 select*。

### 输入框（Input）

<input type="text" class="form-control" placeholder="文本输入">

### 文本框（Textarea）

当您需要进行多行输入的时，则可以使用文本框 textarea。必要时可以改变 rows 属性（较少的行 = 较小的盒子，较多的行 = 较大的盒子）。

<textarea class="form-control" rows="3"></textarea>

### 复选框（Checkbox）和单选框（Radio）

复选框和单选按钮用于让用户从一系列预设置的选项中进行选择。

* 当创建表单时，选择若干个选项使用 *checkbox*。限制用户选择一个选项使用*radio*。
* 对一系列复选框和单选框使用 *.checkbox-inline* 或 *.radio-inline* class，控制它们显示在同一行上。

<label class="**radio-inline**">

<input type="radio" name="optionsRadiosinline" id="optionsRadios3" value="option1" checked> 选项 1 </label>

### 选择框（Select）

当您想让用户从多个选项中进行选择，但是默认情况下只能选择一个选项时，则使用选择框。

* 使用 <select> 展示列表选项，通常是那些用户很熟悉的选择列表，比如州或者数字。
* 使用 *multiple="multiple"* 允许用户选择多个选项。

<select multiple class="form-control">

<option>1</option>

<option>2</option>

<option>3</option>

<option>4</option>

<option>5</option>

</select>

### 静态控件

要在一个水平表单内的表单标签后放置纯文本时，请在 <p> 上使用 class *.form-control-static*。

<p class="form-control-static">email@example.com</p>

## 表单控件状态

除了 *:focus* 状态（即，用户点击 input 或使用 tab 键聚焦到 input 上），Bootstrap 还为禁用的输入框定义了样式，并提供了表单验证的 class。

### 输入框焦点

当输入框 input 接收到 *:focus* 时，输入框的轮廓会被移除，同时应用 *box-shadow*。

### 禁用的输入框 input

如果您想要禁用一个输入框 input，只需要简单地添加 *disabled* 属性，这不仅会禁用输入框，还会改变输入框的样式以及当鼠标的指针悬停在元素上时鼠标指针的样式。

### 禁用的字段集 fieldset

对 <fieldset> 添加 disabled 属性来禁用 <fieldset> 内的所有控件。

### 验证状态

Bootstrap 包含了错误、警告和成功消息的验证样式。只需要对父元素简单地添加适当的 class（*.has-warning、 .has-error 或 .has-success*）即可使用验证状态。

<div class="form-group has-error">

<label class="col-sm-2 control-label" for="inputError">输入错误</label>

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

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

</div>

</div>

## 表单控件大小

您可以分别使用 class *.input-lg* 和 *.col-lg-\** 来设置表单的高度和宽度。

<div class="row">

<div class="col-lg-2">

<input type="text" class="form-control" placeholder=".col-lg-2">

</div>

</div>

## 表单帮助文本

Bootstrap 表单控件可以在输入框 input 上有一个块级帮助文本。为了添加一个占用整个宽度的内容块，请在 <input> 后使用 .help-block。

<span class="help-block">一个较长的帮助文本块，超过一行， 需要扩展到下一行。本实例中的帮助文本总共有两行。</span>

# Bootstrap 按钮

本章将通过实例讲解如何使用 Bootstrap 按钮。任何带有 class **.btn** 的元素都会继承圆角灰色按钮的默认外观。但是 Bootstrap 提供了一些选项来定义按钮的样式，具体如下表所示：

以下样式可用于<a>, <button>, 或 <input> 元素上：

|  |  |
| --- | --- |
| **类** | **描述** |
| .btn | 为按钮添加基本样式 |
| .btn-default | 默认/标准按钮 |
| .btn-primary | 原始按钮样式（未被操作） |
| .btn-success | 表示成功的动作 |
| .btn-info | 该样式可用于要弹出信息的按钮 |
| .btn-warning | 表示需要谨慎操作的按钮 |
| .btn-danger | 表示一个危险动作的按钮操作 |
| .btn-link | 让按钮看起来像个链接 (仍然保留按钮行为) |
| .btn-lg | 制作一个大按钮 |
| .btn-sm | 制作一个小按钮 |
| .btn-xs | 制作一个超小按钮 |
| .btn-block | 块级按钮(拉伸至父元素100%的宽度) |
| .active | 按钮被点击 |
| .disabled | 禁用按钮 |

# Bootstrap 图片

Bootstrap 提供了三个可对图片应用简单样式的 class：

* *.img-rounded*：添加 *border-radius:6px* 来获得图片圆角。
* *.img-circle*：添加 *border-radius:50%* 来让整个图片变成圆形。
* *.img-thumbnail*：添加一些内边距（padding）和一个灰色的边框。

## <img> 类

以下类可用于任何图片中。

|  |  |
| --- | --- |
| **类** | **描述** |
| .img-rounded | 为图片添加圆角 (IE8 不支持) |
| .img-circle | 将图片变为圆形 (IE8 不支持) |
| .img-thumbnail | 缩略图功能 |
| .img-responsive | 图片响应式 (将很好地扩展到父元素) |

## 响应式图片

通过在 <img> 标签添加 .img-responsive 类来让图片支持响应式设计。 图片将很好地扩展到父元素。

.img-responsive 类将 max-width: 100%; 和 height: auto; 样式应用在图片上

<img src="cinqueterre.jpg" class="img-responsive" alt="Cinque Terre">

# Bootstrap 辅助类

本章将讨论 Bootstrap 中的一些可能会派上用场的辅助类。

## 文本

以下不同的类展示了不同的文本颜色。如果文本是个链接鼠标移动到文本上会变暗：

|  |  |
| --- | --- |
| **类** | **描述** |
| .text-muted | "text-muted" 类的文本样式 |
| .text-primary | "text-primary" 类的文本样式 |
| .text-success | "text-success" 类的文本样式 |
| .text-info | "text-info" 类的文本样式 |
| .text-warning | "text-warning" 类的文本样式 |
| .text-danger | "text-danger" 类的文本样式 |

## 背景

以下不同的类展示了不同的背景颜色。 如果文本是个链接鼠标移动到文本上会变暗：

|  |  |
| --- | --- |
| **类** | **描述** |
| .bg-primary | 表格单元格使用了 "bg-primary" 类 |
| .bg-success | 表格单元格使用了 "bg-success" 类 |
| .bg-info | 表格单元格使用了 "bg-info" 类 |
| .bg-warning | 表格单元格使用了 "bg-warning" 类 |
| .bg-danger | 表格单元格使用了 "bg-danger" 类 |

## 其他

|  |  |
| --- | --- |
| **类** | **描述** |
| .pull-left | 元素浮动到左边 |
| .pull-right | 元素浮动到右边 |
| .center-block | 设置元素为 display:block 并居中显示 |
| .clearfix | 清除浮动 |
| .show | 强制元素显示 |
| .hidden | 强制元素隐藏 |
| .sr-only | 除了屏幕阅读器外，其他设备上隐藏元素 |
| .sr-only-focusable | 与 .sr-only 类结合使用，在元素获取焦点时显示(如：键盘操作的用户) |
| .text-hide | 将页面元素所包含的文本内容替换为背景图 |
| .close | 显示关闭按钮 |
| .caret | 显示下拉式功能 |

### 关闭图标

使用通用的关闭图标来关闭模态框和警告框。使用 class **close** 得到关闭图标。

### 插入符

使用插入符表示下拉功能和方向。使用带有 class **caret** 的 <span> 元素得到该功能。

### 快速浮动

您可以分别使用 class **pull-left** 或 **pull-right** 来把元素向左或向右浮动。下面的实例演示了这点。

如需对齐导航栏中的组件，请使用 **.navbar-left** 或 **.navbar-right** 代替.

### 清除浮动

如需清除元素的浮动，请使用 **.clearfix** class。

### 显示和隐藏内容

您可以通过使用 class **.show** 和 **.hidden** 来强行设置元素显示或隐藏（包括屏幕阅读器）。