BootStrap

讲师：莫凡

# 课程回顾

# 课程介绍

* 1.栅格系统
* 2.表格
* 3.表单
* 4.导航栏
* 5.轮播

1. BootStrap概述
   1. 什么是BootStrap



BootStrap设计出响应式页面,由它设计页面可以在手机,PAD,PC都可以直接访问.

* 1. 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。

1. BootStrap起步
   1. 下载



* 1. 引入的文件



* 1. 添加一个meta标签

移动设备优先是 **Bootstrap 3 的最显著的变化。**

在之前的 Bootstrap 版本中（直到 2.x），您需要手动引用另一个 CSS，才能让整个项目友好的支持移动设备。现在不一样了，Bootstrap 3 默认的 CSS 本身就对移动设备友好支持。Bootstrap 3 的设计目标是移动设备优先，然后才是桌面设备。这实际上是一个非常及时的转变，因为现在越来越多的用户使用移动设备。为了让 Bootstrap 开发的网站对移动设备友好，确保适当的绘制和触屏缩放，需要在网页的 head 之中添加 viewport meta 标签，如下所示：

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

width 属性**控制设备的宽度**。假设您的网站将被带有不同屏幕分辨率的设备浏览，那么将它设置为 device-width 可以确保它能正确呈现在不同设备上。

initial-scale=1.0 确保**网页加载时，以 1:1 的比例呈现**，不会有任何的缩放。

在移动设备浏览器上，通过为 viewport meta 标签添加 user-scalable=no 可以禁用其缩放（zooming）功能。

通常情况下，maximum-scale=1.0 与 user-scalable=no 一起使用。这样禁用缩放功能后，用户只能滚动屏幕，就能让您的网站看上去更像原生应用的感觉。

注意，这种方式我们并不推荐所有网站使用，还是要看您自己的情况而定！

<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">

1. 布局容器

Bootstrap 需要为页面内容和栅格系统包裹一个 .container 容器。我们提供了两个作此用处的类。注意，由于 padding 等属性的原因，这两种容器类不能互相嵌套。

.container 类用于固定宽度并支持响应式布局的容器。

<div class="container">

...

</div>

.container-fluid 类用于 100% 宽度，占据全部视口（viewport）的容器。

<div class="container-fluid">

...

</div>

1. 栅格系统

Bootstrap 提供了一套**响应式、**移动设备优先的流式栅格系统，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. 什么是网格（Grid）？

摘自维基百科：

在平面设计中，网格是一种由一系列用于组织内容的相交的直线（垂直的、水平的）组成的结构（通常是二维的）。它广泛应用于打印设计中的设计布局和内容结构。在网页设计中，它是一种用于快速创建一致的布局和有效地使用 HTML 和 CSS 的方法。

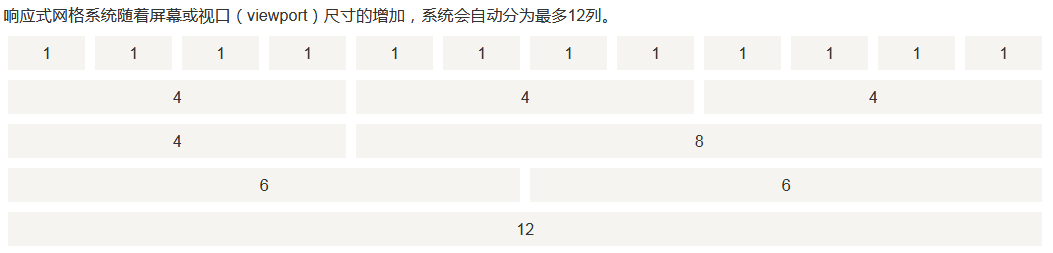
简单地说，网页设计中的网格用于组织内容，让网站易于浏览，并降低用户端的负载。

* 1. 什么是 Bootstrap 网格系统（Grid System）？

Bootstrap 官方文档中有关网格系统的描述：

Bootstrap 包含了一个响应式的、移动设备优先的、不固定的网格系统，可以随着设备或视口大小的增加而适当地扩展到 12 列。它包含了用于简单的布局选项的预定义类，也包含了用于生成更多语义布局的功能强大的混合类。

让我们来理解一下上面的语句。Bootstrap 3 是移动设备优先的，在这个意义上，Bootstrap 代码从小屏幕设备（比如移动设备、平板电脑）开始，然后扩展到大屏幕设备（比如笔记本电脑、台式电脑）上的组件和网格。



* 1. Bootstrap 网格系统的工作原理

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）。

使用行来创建列的水平组。

内容应该放置在列内，且唯有列可以是行的直接子元素。

预定义的网格类，比如 .row 和 .col-xs-4，可用于快速创建网格布局。LESS 混合类可用于更多语义布局。

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。



* 1. 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>

* 1. 代码示例

<div class="container">

<div class="col-md-3">111</div>

<div class="col-md-3">111</div>

<div class="col-md-3">111</div>

<div class="col-md-3">111</div>

<div class="col-md-8">

<div class="row">

<div class="col-md-3">222</div>

<div class="col-md-3">222</div>

<div class="col-md-3">222</div>

<div class="col-md-3">222</div>

</div>

</div>

<div class="col-md-2">111</div>

<div class="col-md-1">111</div>

<div class="col-md-1">111</div>

</div>

<div class="container">

<div class="row">

<div class="col-md-3 col-sm-6 col-xs-12">

<h1>关于我们</h1>

<p>关于我们关于我们关于我们关于我们关于我们</p>

</div>

<div class="col-md-3 col-sm-6 col-xs-12">

<h1>友情链接</h1>

<p>关于我们关于我们关于我们关于我们关于我们</p>

</div>

<div class="col-md-3 col-sm-6 col-xs-12">

<h1>社会动态</h1>

<p>关于我们关于我们关于我们关于我们关于我们</p>

</div>

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

<h1>人间冷暖</h1>

<p>关于我们关于我们关于我们关于我们关于我们</p>

</div>

</div>

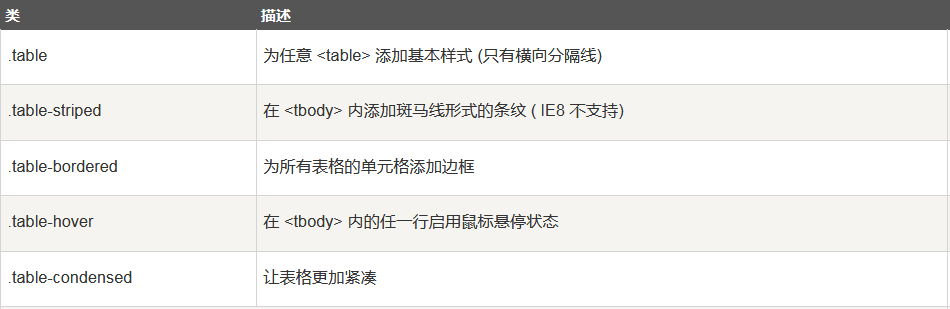
</div>

1. 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：



* 1. 可用于表格的样式



* 1. 可用于tr/th/td的样式



* 1. 响应式表格

通过把任意的 .table 包在 **.table-responsive** class 内，可以让表格水平滚动以适应小型设备（小于 768px）。当在大于 768px 宽的大型设备上查看时，将看不到任何的差别。

<div class="table-responsive">

<table class="table">

<caption>响应式表格布局</caption>

<thead>

<tr>

<th>产品</th>

<th>付款日期</th>

<th>状态</th></tr>

</thead>

<tbody>

<tr>

<td>产品1</td>

<td>23/11/2013</td>

<td>待发货</td></tr>

<tr>

<td>产品2</td>

<td>10/11/2013</td>

<td>发货中</td></tr>

<tr>

<td>产品3</td>

<td>20/10/2013</td>

<td>待确认</td></tr>

<tr>

<td>产品4</td>

<td>20/10/2013</td>

<td>已退货</td></tr>

</tbody>

</table>

</div>

1. 表单

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

Bootstrap 支持最常见的表单控件，主要是 input、textarea、checkbox、radio 和 select。

* 1. 垂直表单

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：

* + - 1. 向父 <form> 元素添加 role="form"。
      2. 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。
      3. 向所有的文本元素 <input>、<textarea> 和 <select> 添加 class ="form-control" 。

代码示例：

<form role="form">

<div class="form-group">

<label for="name">名称</label>

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

</div>

<div class="form-group">

<label for="inputfile">文件输入</label>

<input type="file" id="inputfile">

<p class="help-block">这里是块级帮助文本的实例。</p>

</div>

<div class="checkbox">

<label>

<input type="checkbox">请打勾

</label>

</div>

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

</form>

* 1. 内联表单

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

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

<div class="form-group">

<label class="sr-only" for="name">名称</label>

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

</div>

<div class="form-group">

<label class="sr-only" for="inputfile">文件输入</label>

<input type="file" id="inputfile">

</div>

<div class="checkbox">

<label>

<input type="checkbox">请打勾

</label>

</div>

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

</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。

* 1. 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：

* + - 1. 向父 <form> 元素添加 class .**form-horizontal**。
      2. 把标签和控件放在一个带有 class .form-group 的 <div> 中。
      3. 向标签添加 class .control-label。

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

<div class="form-group">

<label for="firstname" class="col-sm-2 control-label">名字</label>

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

<input type="text" class="form-control" id="firstname" placeholder="请输入名字">

</div>

</div>

<div class="form-group">

<label for="lastname" class="col-sm-2 control-label">姓</label>

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

<input type="text" class="form-control" id="lastname" placeholder="请输入姓">

</div>

</div>

<div class="form-group">

<div class="col-sm-offset-2 col-sm-10">

<div class="checkbox">

<label>

<input type="checkbox">请记住我

</label>

</div>

</div>

</div>

<div class="form-group">

<div class="col-sm-offset-2 col-sm-10">

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

</div>

</div>

</form>

1. 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。

* 1. 默认导航栏

创建一个默认的导航栏的步骤如下：

* + - 1. 向 <nav> 标签添加 class .navbar、.navbar-default。
      2. 向上面的元素添加 role="navigation"，有助于增加可访问性。
      3. 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。
      4. 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。

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

<div class="container-fluid">

<div class="navbar-header">

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

</div>

</nav>

* 1. 响应式导航栏

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 class .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 <span> 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。

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

<div class="container-fluid">

<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="#">教程</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>

</div>

</nav>

1. 轮播

Bootstrap 轮播（Carousel）插件是一种灵活的响应式的向站点添加滑块的方式。除此之外，内容也是足够灵活的，可以是图像、内嵌框架、视频或者其他您想要放置的任何类型的内容。

* 1. 基本轮播

<div id="myCarousel" class="carousel slide">

<!-- 轮播（Carousel）指标 -->

<ol class="carousel-indicators">

<li data-target="#myCarousel" data-slide-to="0" class="active"></li>

<li data-target="#myCarousel" data-slide-to="1"></li>

<li data-target="#myCarousel" data-slide-to="2"></li>

</ol>

<!-- 轮播（Carousel）项目 -->

<div class="carousel-inner">

<div class="item active">

<img src="/wp-content/uploads/2014/07/slide1.png" alt="First slide">

</div>

<div class="item">

<img src="/wp-content/uploads/2014/07/slide2.png" alt="Second slide">

</div>

<div class="item">

<img src="/wp-content/uploads/2014/07/slide3.png" alt="Third slide">

</div>

</div>

<!-- 轮播（Carousel）导航 -->

<a class="carousel-control left" href="#myCarousel"

data-slide="prev"> <span \_ngcontent-c3="" aria-hidden="true" class="glyphicon glyphicon-chevron-right"></span></a>

<a class="carousel-control right" href="#myCarousel"

data-slide="next">&rsaquo;</a>

</div>

* 1. 带标题的轮播

可以通过 .item 内的 .carousel-caption 元素向幻灯片添加标题。只需要在该处放置任何可选的 HTML 即可，它会自动对齐并格式化。

<div id="myCarousel" class="carousel slide">

<!-- 轮播（Carousel）指标 -->

<ol class="carousel-indicators">

<li data-target="#myCarousel" data-slide-to="0" class="active"></li>

<li data-target="#myCarousel" data-slide-to="1"></li>

<li data-target="#myCarousel" data-slide-to="2"></li>

</ol>

<!-- 轮播（Carousel）项目 -->

<div class="carousel-inner">

<div class="item active">

<img src="/wp-content/uploads/2014/07/slide1.png" alt="First slide">

<div class="carousel-caption">标题 1</div>

</div>

<div class="item">

<img src="/wp-content/uploads/2014/07/slide2.png" alt="Second slide">

<div class="carousel-caption">标题 2</div>

</div>

<div class="item">

<img src="/wp-content/uploads/2014/07/slide3.png" alt="Third slide">

<div class="carousel-caption">标题 3</div>

</div>

</div>

<!-- 轮播（Carousel）导航 -->

<a class="left carousel-control" href="#myCarousel" role="button" data-slide="prev">

<span class="glyphicon glyphicon-chevron-left" aria-hidden="true"></span>

<span class="sr-only">Previous</span>

</a>

<a class="right carousel-control" href="#myCarousel" role="button" data-slide="next">

<span class="glyphicon glyphicon-chevron-right" aria-hidden="true"></span>

<span class="sr-only">Next</span>

</a>

</div>

1. 课程总结
   1. 重点
   2. 难点
   3. 如何掌握？
      * 1. 勤加练习
        2. 学会看说明手册
   4. 排错技巧（技巧）
2. 课后练习
3. 面试题
4. 扩展知识或课外阅读推荐（可选）
   1. 扩展知识
   2. 课外阅读