

Bootstrap基础入门

今日内容介绍

- ◆ 使用 bootstrap 重写首页

今日内容学习目标

- ◆ 学会通过官方提供 demo 实例，完成自己需要的功能。
- ◆ 能够从已有 html 文档中，找到将要修改的位置，并进行简单调整

第1章 内容回顾

第2章 案例：重写首页

2.1 案例介绍

将使用 Bootstrap 重写首页，整个案例中将使用到 Bootstrap 各种模块，为了方便编程，将采用拆分的原则，各个模块单独编写，最后组合。

2.2 相关技术介绍

2.2.1 BootStrap 概述

2.2.1.1 什么是 BootStrap

- Bootstrap，基于 HTML、CSS、JAVASCRIPT 的前端框架（半成品）。其预定义一套 CSS 样式和与样式对应的 jQuery 代码，我们只需要提供固定 HTML 结构，添加固定的 class 样式，就可以完成指定效果的实现。
- Bootstrap 在 jQuery 的基础工作，可以理解 Bootstrap 就是 jQuery 的一个插件。
- Bootstrap 使得 Web 开发更加快捷，代码优雅，美观大方。
- 由 Twitter 公司的设计师 Mark Otto 和 Jacob Thornton 合作开发。
- Bootstrap 基础入门使用的都是自带 CSS 样式，高级开发中需要使用 HTML5、CSS3、动态 CSS 语言 Less 进行自定义开发，JavaEE 课程中学习时“基础入门”。
- 国内一些移动开发者较为熟悉的框架，如 WeX5 前端开源框架等，也是基于 Bootstrap 源码进行性能优化而来
- 中文官网：<http://www.bootcss.com/>

2.2.1.2 什么是响应式布局

- 响应式布局：一个网站能够兼容多个终端(手机、iPad 等)，而不需要为每个终端做一个特定的版本。此概念是为解决移动互联网浏览而诞生的。
- 响应式布局可以为不同终端的用户提供更加舒适的界面和更好的用户体验，而且随着目前大屏幕移动设备的普及，用“大势所趋”来形容也不为过。随着越来越多的设计师采用这个技术，我们不仅看到很多的创新，还看到了一些成形的模式。
- Bootstrap 就是响应式布局最成功的实现，为了兼容不同的浏览器采用 jQuery，为了适配不同的终端采用 CSS3 Media Query（媒体查询）

2.2.2 环境搭建

2.2.2.1 下载

- 中文官网地址：<http://d.bootcss.com/bootstrap-3.3.5.zip>
- GIT 地址：<https://github.com/twbs/bootstrap/archive/v3.3.6.zip>

用于生产环境的 Bootstrap

编译并压缩后的 CSS、JavaScript 和字体文件。不包含文档和源码文件。

[下载 Bootstrap](#)

Bootstrap 源码

Less、JavaScript 和字体文件的源码，并且带有文档。需要 Less 编译器和一些设置工作。

[下载源码](#)

Sass

这是 Bootstrap 从 Less 到 Sass 的源码移植项目，用于快速地在 Rails、Compass 或只针对 Sass 的项目中引入。

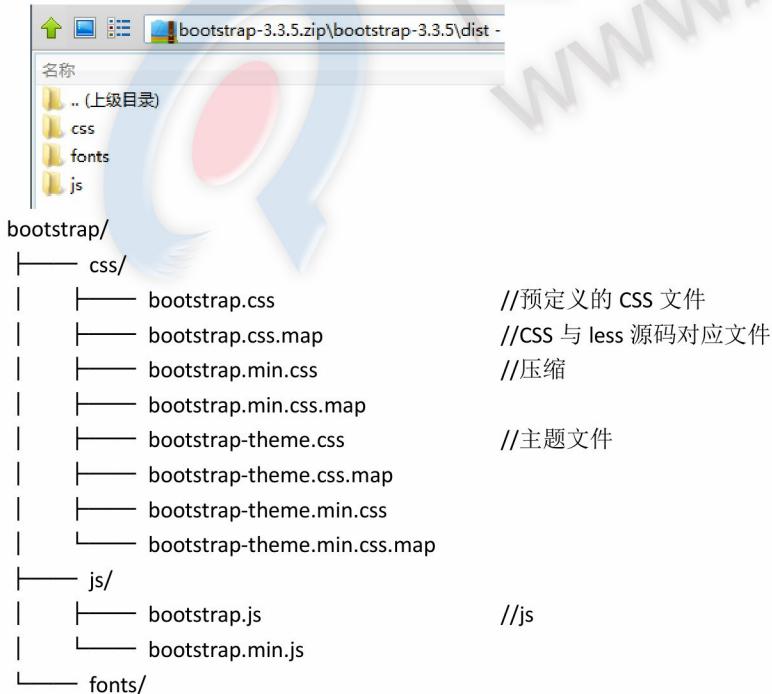
[下载 Sass 项目](#)

2.2.2.2 目录结构



2.2.2.3 内容结构

- 发布版，及开始使用到的 Bootstrap 内容结构



```
|---- glyphicons-halflings-regular.eot      //字体 (字体图标)
|---- glyphicons-halflings-regular.svg
|---- glyphicons-halflings-regular.ttf
|---- glyphicons-halflings-regular.woff
|---- glyphicons-halflings-regular.woff2
```

2.2.2.4 简洁模板

```
<!DOCTYPE html>
<html lang="zh-CN">

    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width,initial-scale=1" />
        <title>Bootstrap 模板</title>

        <link href="../lib/bootstrap/css/bootstrap.min.css" rel="stylesheet">
        <script src="../lib/jquery/jquery-1.11.0.js"></script>
        <script src="../lib/bootstrap/js/bootstrap.min.js"></script>

    </head>

    <body>
        <h1>你好，世界！</h1>
    </body>

</html>
```

批注 [It1]: HTML5 约束 (固定值)

批注 [It2]: 建议编写, 生命语言, 暂没有发现用途

批注 [It3]: 响应式开发必须使用, 且必须在<head>前三行

批注 [It4]: 预定义的 CSS 样式

批注 [It5]: 基于 jQuery

批注 [It6]: Bootstrap 类库

2.2.2.5 完整模板 (了解)

- 参考文档: <http://v3.bootcss.com/getting-started/#template>

```
<!DOCTYPE html>
<html lang="zh-CN">
    <head>
        <meta charset="utf-8">
        <!--此属性为文档兼容(compatible)模式声明, 表示使用 IE 浏览器的最新渲染模式--&gt;
        &lt;meta http-equiv="X-UA-Compatible" content="IE=edge"&gt;
        &lt;!--
            视口: 用于设置移动浏览器显示效果。
            视口的作用: 在移动浏览器中, 当页面宽度超出设备(device), 浏览器内部虚拟的一个页面
        --&gt;</pre>
```

容器，将页面容器缩放到设备这么大，然后展示

取值：

width=device-width，视口的宽度，大多手机浏览器视口的宽度是 980
device-width 表示采用设备的宽度
initial-scale=1 初始化缩放级别，取值：1-5
minimum-scale=1 最小缩放级别
maximum-scale=1 最大缩放级别
user-scalable=no 禁用缩放
如果设置“user-scalable=no”，则“minimum-scale”和“maximum-scale”无效

下面实例表示的意思：

根据设置确定视口宽度，初始化的缩放比例是 2.5，最小缩放比例 1.5，最大缩放比例为 3，允许在移动设置缩放

常用值：

```
<meta name="viewport" content="width=device-width, initial-scale=1">
-->
<meta name="viewport" content="width=device-width,initial-scale=2.5,minimum-scale=1.5,maximum-scale=3,user-scalable=yes" />
<!-- 上述 3 个 meta 标签必须放在最前面，任何其他内容都必须跟随其后！ --&gt;

&lt;title&gt;Bootstrap 模板&lt;/title&gt;

&lt!-- Bootstrap --&gt;
&lt;link href="../lib/bootstrap/css/bootstrap.min.css" rel="stylesheet"&gt;

<!-- 使 IE8 支持 HTML5 元素特性和 CSS3 媒体查询
注意：respond.js 不能本地运行 (file://)，必须放置在 web 服务器 (http://)，
暂时不用掌握)
--&gt;
&lt;!--[if lt IE 9]&gt;
&lt;script src="../lib/html5shiv/html5shiv.min.js"&gt;&lt;/script&gt;
&lt;script src="../lib/respond/respond.min.js"&gt;&lt;/script&gt;
&lt;![endif]--&gt;
&lt;/head&gt;
&lt;body&gt;
&lt;!--正文从此处开始--&gt;
&lt;h1&gt;你好，世界！&lt;/h1&gt;</pre>
```

<!--前端开发中建议：网站优化时，除了立即需要工作的 js 存放在 header 外，将大部分 js 文件放置在页面的末尾-->

```
<!-- Bootstrap 必须在 jQuery 的基础上工作-->
<script src="../lib/jquery/jquery-1.11.0.js"></script>
<!-- Bootstrap 核心包 -->
<script src="../lib/bootstrap/js/bootstrap.min.js"></script>
```

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

2.3 重写首页之 topbar

2.3.1 案例相关技术

2.3.1.1 布局容器

- 帮助手册: <http://v3.bootcss.com/css/#overview-container>
- Bootstrap 需要为页面内容和栅格系统包裹一个 `.container` 容器。提供的两个容器如下:
 - `.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>
```

- 例如:

```
<!--居中显示，两边有留白-->  
<div class="container" style="border:1px solid #f00; height:100px;"></div>  
<!--整个宽度-->  
<div class="container-fluid" style="border:1px solid #f00; height:100px;"></div>
```

2.3.1.2 栅格

- 帮助手册: <http://v3.bootcss.com/css/#grid-options>
- 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。
- 栅格特点
 - “行（row）”必须包含在 `.container`（固定宽度）或 `.container-fluid`（100% 宽度）中
 - “列（column）”可以作为行（row）的直接子元素。
 - 行使用的样式“`.row`”，列使用样式“`col-*-*`”内容应当放置于“列（column）”内
 - 列大于 12 时，将另起一行排列
 - 栅格类适用于与屏幕宽度大于或等于分界点大小的设备，并且针对小屏幕设备覆盖栅格类。

● 棚格参数：“**col-*-***”

	超小屏幕 手机 (<768px)	小屏幕 平板 (≥768px)	中等屏幕 桌面显示器 (≥992px)	大屏幕 大桌面显示器 (≥1200px)	
栅格系统行为	总是水平排列	开始是堆叠在一起的，当大于这些阈值时将变为水平排列			
.container 最大宽度	None (自动)	750px	970px	1170px	
类前缀	.col-xs-	.col-sm-	.col-md-	.col-lg-	
列 (column) 数	12				
最大列 (column) 宽	自动	~62px	~81px	~97px	
槽 (gutter) 宽	30px (每列左右均有 15px)				
可嵌套	是				
偏移 (Offsets)	是				
列排序	是				

large : lg

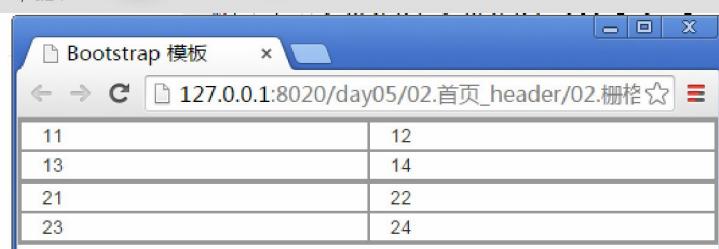
medium : md

small: sm : sm

x small : xs

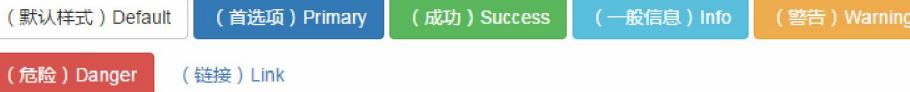
● 例如：

```
<div class="container">
    <div class="row">
        <div class="col-md-3 col-xs-6">11</div>
        <div class="col-md-3 col-xs-6">12</div>
        <div class="col-md-3 col-xs-6">13</div>
        <div class="col-md-3 col-xs-6">14</div>
    </div>
    <div class="row">
        <div class="col-md-3 col-xs-6">21</div>
        <div class="col-md-3 col-xs-6">22</div>
        <div class="col-md-3 col-xs-6">23</div>
        <div class="col-md-3 col-xs-6">24</div>
    </div>
</div>
```



2.3.1.3 按钮

- 帮助手册: <http://v3.bootcss.com/css/#buttons-options>



```
<button type="button" class="btn btn-default"> (默认样式) Default</button>
<button type="button" class="btn btn-primary"> (首选项) Primary</button>
<button type="button" class="btn btn-success"> (成功) Success</button>
<button type="button" class="btn btn-info"> (一般信息) Info</button>
<button type="button" class="btn btn-warning"> (警告) Warning</button>
<button type="button" class="btn btn-danger"> (危险) Danger</button>
<button type="button" class="btn btn-link"> (链接) Link</button>
```

- .btn-lg、.btn-sm 或 .btn-xs 可以设置按钮的不同尺寸
- .active 类设置按钮激活状态，其表现为被按压下去（底色更深、边框变色更深、向内投射阴影）

2.3.1.4 响应式工具

- 帮助手册: <http://v3.bootcss.com/css/#responsive-utilities-classes>

	超小屏幕 手机 (<768px)	小屏幕 平板 (≥768px)	中等屏幕 桌面 (≥992px)	大屏幕 桌面 (≥1200px)
.visible-xs-*	可见	隐藏	隐藏	隐藏
.visible-sm-*	隐藏	可见	隐藏	隐藏
.visible-md-*	隐藏	隐藏	可见	隐藏
.visible-lg-*	隐藏	隐藏	隐藏	可见
.hidden-xs	隐藏	可见	可见	可见
.hidden-sm	可见	隐藏	可见	可见
.hidden-md	可见	可见	隐藏	可见
.hidden-lg	可见	可见	可见	隐藏

- 例如:

```
<!--
    设置 one div，中等屏幕和超小屏幕显示
    设置 two div，小屏幕和超大屏幕隐藏
-->


one</div>


two</div>


```

2.3.2 案例实现

```
<!--
1.整个 topbar 划分比例: 1:2:1
2.中间区域只在“大屏幕”和“中等屏幕”显示
3.整个区域在“超小屏幕”英寸

-->










免费注册
登录
购物车


</div>
```

2.4 重写首页之导航

2.4.1 案例分析

- Bootstrap 已经提供了导航的完整实例，通常情况下，只需要进行简单修改即可。
- 帮助文档：<http://v3.bootcss.com/components/#navbar>
- 反色导航条：<http://v3.bootcss.com/components/#navbar-inverted>

2.4.2 案例实现

```
<!--
1.大屏幕显示所有分类
2.中等屏幕隐藏部分分类，提供“更多”
3.超小屏幕隐藏所有分类，以“汉堡按钮”显示

-->
```

```
<div class="container">
    <nav class="navbar navbar-default navbar-inverse">
        <div class="container-fluid">
            <!-- Brand and toggle get grouped for better mobile display -->
            <div class="navbar-header">
                <button type="button" class="navbar-toggle collapsed"
data-toggle="collapse" data-target="#bs-example-navbar-collapse-1"
aria-expanded="false">
                    <span class="sr-only">汉堡按钮</span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                <a class="navbar-brand" href="#">首页</a>
            </div>

            <!-- Collect the nav links, forms, and other content for toggling -->
            <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
                <ul class="nav navbar-nav">
                    <li class="active"><a href="#">电 脑 办 公 <span
class="sr-only">(current)</span></a></li>
                    <li><a href="#">电脑办公</a></li>
                    <li><a href="#">电脑办公</a></li>
                    <li class="hidden-sm"><a href="#">电脑办公</a></li>
                    <li class="dropdown visible-sm">
                        <a href="#" class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="false">更 多 <span
class="caret"></span></a>
                        <ul class="dropdown-menu">
                            <li><a href="#">电脑办公</a></li>
                            <li><a href="#">电脑办公</a></li>
                            <li><a href="#">电脑办公</a></li>
                            <li role="separator" class="divider"></li>
                            <li><a href="#">电脑办公</a></li>
                            <li role="separator" class="divider"></li>
                            <li><a href="#">电脑办公</a></li>
                        </ul>
                    </li>
                </ul>
                <form class="navbar-form navbar-left navbar-right" role="search">
                    <div class="form-group">
                        <input type="text" class="form-control" placeholder="Search">
                    </div>
                    <button type="submit" class="btn btn-default">
                </div>
            </div>
        </div>
    </nav>
</div>
```

```
hidden-xs">Submit</button>
    </form>
</div><!-- /.navbar-collapse -->
</div><!-- /.container-fluid -->
</nav>
</div>
```

2.5 重写首页之轮播图

2.5.1 案例分析

- Bootstrap 已经提供轮播图的完整案例，此处我们也是进行修改即可
- 参考文档：<http://v3.bootcss.com/javascript/#carousel>

2.5.2 案例实现

```
<div class="container">
    <div id="index_carousel" class="carousel slide" data-ride="carousel">
        <!-- 指示器 Indicators -->
        <ol class="carousel-indicators">
            <li data-target="#index_carousel" data-slide-to="0" class="active"></li>
            <li data-target="#index_carousel" data-slide-to="1"></li>
            <li data-target="#index_carousel" data-slide-to="2"></li>
        </ol>

        <!-- 轮播展示
             * item 表示一个图片，与“指示器”li 的个数保持一致
        -->
        <div class="carousel-inner" role="listbox">
            <div class="item active">
                
            </div>
            <div class="item">
                
            </div>
            <div class="item">
                
            </div>
        </div>
    </div>
```

```
</div>

<!-- 左右控制区 Controls
 * href 用于确定点击触发的那个轮播图
-->
<a class="left carousel-control" href="#index_carousel" role="button"
data-slide="prev">
    <span class="glyphicon glyphicon-chevron-left" aria-hidden="true"></span>
    <span class="sr-only">前一张</span>
</a>
<a class="right carousel-control" href="#index_carousel" role="button"
data-slide="next">
    <span class="glyphicon glyphicon-chevron-right" aria-hidden="true"></span>
    <span class="sr-only">下一张</span>
</a>
</div>

</div>
```

2.6 重写首页之热卖商品

2.6.1 案例分析

“热卖商品”模块是对栅格系统的再应用。我们将用已有的知识编写出响应式页面。

2.6.2 案例实现

```
<div class="container">
    <div class="row">
        <div class="col-md-12">
            <h2>热卖商品 </h2>
        </div>
    </div>
    <div class="row">
        <div class="col-md-2 col-sm-4 hidden-xs">
            
        </div>
        <div class="col-md-10 col-sm-8">
            <div class="col-md-6 col-sm-12" style="height:200px;">
                
</div>
<div class="col-md-2 product col-sm-4 col-xs-6">
    
    <p>电器</p>
    <p style="font-size: #f00;">¥998</p>
</div>
<div class="col-md-2 product col-sm-4 col-xs-6">
    
    <p>电器</p>
    <p style="font-size: #f00;">¥998</p>
</div>
<div class="col-md-2 product col-sm-4 col-xs-6">
    
    <p>电器</p>
    <p style="font-size: #f00;">¥998</p>
</div>

<div class="col-md-2 product col-sm-4 col-xs-6">
    
    <p>电器</p>
    <p style="font-size: #f00;">¥998</p>
</div>
<div class="col-md-2 product col-sm-4 col-xs-6">
    
    <p>电器</p>
    <p style="font-size: #f00;">¥998</p>
</div>
<div class="col-md-2 product col-sm-4 col-xs-6">
    
    <p>电器</p>
    <p style="font-size: #f00;">¥998</p>
</div>
<div class="col-md-2 product col-sm-4 col-xs-6">
    
    <p>电器</p>
    <p style="font-size: #f00;">¥998</p>
</div>
<div class="col-md-2 product col-sm-4 col-xs-6">
    
    <p>电器</p>
    <p style="font-size: #f00;">¥998</p>
</div>
<div class="col-md-2 product col-sm-4 col-xs-6">
```

```

<p>电器</p>
<p style="font-size: #f00;">¥998</p>
</div>

</div>
</div>

</div>
```

2.7 重写首页之 footer

2.7.1 案例分析

通过分析，该模块需要使用栅格划分上下两行，第二行的文字信息准备使用“列表”，内容的居中将提供两种方案：文字居中，栅格列偏移



参考文档-文本对齐：<http://v3.bootcss.com/css/#type-alignment>

参考文档-列表：<http://v3.bootcss.com/css/#type-lists>

参考文档-栅格列偏移：<http://v3.bootcss.com/css/#grid-offsetting>

2.7.2 案例实现

```
<div class="container">
<div class="row">
    
</div>
<div class="row">
    <!--可以使用 text-center 文本居中-->
    <div class="col-md-8 col-md-offset-3">
        <ul class="list-inline">
            <li><a href="#">关于我们</a></li>
            <li><a href="#">联系我们</a></li>
            <li><a href="#">招贤纳士</a></li>
            <li><a href="#">法律声明</a></li>
            <li><a href="#">友情链接</a></li>
            <li><a href="#">支付方式</a></li>
            <li><a href="#">配送方式</a></li>
        </ul>
    </div>
</div>
```

```
<li><a href="#">免责声明</a></li>
<li><a href="#">广告声明</a></li>
</ul>
</div>

</div>
```

总结



传智播客
www.itcast.cn