

HTML5程序设计基础

第21章 Bootstrap概览及栅格系统

主要内容

01

初识 Bootstrap

02

Bootstrap 环境安装

03

Bootstrap CSS 概览

04

Bootstrap 栅格系统


01

初识 Bootstrap

Bootstrap

- Bootstrap 是在2011年，由twitter的“一小撮”工程师为了提高内部的协调性和工作效率，用业余时间为他们产品构建的一套易用、优雅、灵活、可扩展的**前端工具集**。
- 简单灵活，用于**快速开发** Web 应用程序和网站的前端框架。

基于HTML、CSS、JavaScript



响应式

Bootstrap版本

- Bootstrap 是全球最受欢迎的前端组件库，用于开发响应式布局、移动设备优先的 WEB 项目。
- Bootstrap4 是目前的最新版本，是一套用于 HTML、CSS 和 JS 开发的开源工具集。
- Bootstrap4与 3 相比拥有更多的具体的类及组件。
- Bootstrap.min.css 的体积减少了40%以上。
- 放弃对 IE8 和 iOS6 的支持，仅支持 IE9 以上及 iOS 7 以上版本的浏览器。

V4.1.3 V3.3.7 V2.3.2

Bootstrap的优势

- **移动设备优先**：自 Bootstrap 3 起，框架包含了贯穿于整个库的移动设备优先的样式。
- **浏览器支持**：所有的主流浏览器都支持 Bootstrap。
- **容易上手**：只要具备 HTML 和 CSS 的知识，就可以开始学习。
- **响应式设计**：响应式 CSS 能够自适应于台式机、平板电脑和手机。
- 包含了功能强大的内置**组件**，易于定制。
- **开源**。

缺点：不方便修改样式，对细节的不好把握



Bootstrap

就业前景

职位职能: 高级软件工程师 软件

职位描述:

岗位职责:

- 1、负责公司业务项目的软件开发;
- 2、负责对项目进行实施, 测试;
- 3、定期进行项目维护和升级;
- 4、负责编写优质的产品相关手册;

任职要求:

- 1、计算机及相关专业毕业, 专科以
- 2、三年以上ASP.NET开发经验, 精
- 3、精通SQLServer、Oracle任意一
- 4、精通ASP.NET MVC、Bootstrap
- 5、熟练掌握HTML5+CSS3网页设计
- 6、有独立分析需求、设计用例模型
- 7、具有良好的沟通能力、团队合作
- 8、有信息发布、门禁系统、通讯软

职位标签: Java JavaScript Ajax HTML5 DIV CSS

职位职能: 高级软件工程师

职位描述:

工作内容:

1. 参与产品线Web的前后端开发
2. 与业务、产品紧密合作, 分析需求, 完成系统的设计、开发、扩展或代码重构
3. 设计和规划所辖系统的整体架构和技术规范, 并在日常开发实践中贯彻执行
4. 负责技术难点的攻关和关键系统的设计和编码
5. 对个人质量和进度负责, 帮助提升开发人员技能水平和开发效率

岗位要求:

1. 熟悉DIV+CSS, HTML5, CSS3, JS, Ajax, Json, Bootstrap等前端技术
2. 熟练使用JQuery, Angular等Javascript框架, 对Javascript性能优化, 安全性有丰富经验
3. 对HTML、W3C标准有较深的理解, 精通各种主流浏览器兼容性问题的解决
4. 精通Spring Foundation, Maven, JPA, Restful API, Nio等J2EE技术
5. 具备面向对象的设计思路和经验, 熟悉软件开发流程, 精通UML语言、设计模式
6. 精通数据库设计, 深入了解性能调优
7. 有大型互联网站点或高并发、高性能Web项目经验
8. 思路清晰, 具备良好的沟通能力和理解能力, 责任心强, 具有一定的团队合作精神, 认真负责
9. 有良好的英文资料的书写、阅读及英文口语表达能力
10. 有多语言站点、分布式应用开发经验、有大型网站经验优先

Bootstrap网站

官网：<https://getbootstrap.com/>

中文网：<http://www.bootcss.com/>



Bootstrap网站实例

<http://expo.bootcss.com/>



Bootstrap包含的内容

- **基本结构**：提供了一个带有网格系统、链接样式、背景的基本结构。
- **CSS**：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。
- **组件**：可重用的常见界面组件，用于创建进度条、下拉菜单、导航、按钮组、分页等等。
- **JavaScript 插件**：和组件类似，用来实现提示、弹出框、模态对话框、轮播等具有交互性的组件。
- **定制**：可以定制 Bootstrap 的组件、LESS 变量和 jQuery 插件来得到自定义版本。

02

Bootstrap环境安装

下载Bootstrap

资源下载：<http://v3.bootcss.com/getting-started/>

编译好的文件

- CSS、JS和图片文件

源码

- Bootstrap源码及文档

下载Bootstrap

1. 下载Bootstrap

编译并压缩后的 CSS、JavaScript 和字体文件

下载源码

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Bootstrap

2. 使用 BootCDN 提供的免费 CDN 加速服务（同时支持 http 和 https 协议）

```
<!-- jQuery文件。务必在bootstrap.min.js 之前引入 -->
<script
src="http://cdn.static.runoob.com/libs/jquery/2.1.1/jquery.min.js"></script>
<!-- 最新版本的 Bootstrap 核心 CSS 文件 -->
<link rel="stylesheet"
href="https://cdn.bootcss.com/bootstrap/3.3.7/css/bootstrap.min.css">
<!-- 可选的 Bootstrap 主题文件（一般不用引入） -->
<link rel="stylesheet"
href="https://cdn.bootcss.com/bootstrap/3.3.7/css/bootstrap-theme.min.css">
<!-- 最新的 Bootstrap 核心 JavaScript 文件 -->
<script
src="https://cdn.bootcss.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
```


Bootstrap文件结构

Bootstrap 的所有插件都依赖 jQuery，因此 jQuery 必须在 Bootstrap 之前引入！

预编译Bootstrap的文件/目录结构：

```
bootstrap/
├── css/
│   ├── bootstrap.css
│   ├── bootstrap.css.map
│   ├── bootstrap.min.css
│   ├── bootstrap-theme.css
│   ├── bootstrap-theme.css.map
│   └── bootstrap-theme.min.css
├── js/
│   ├── bootstrap.js
│   └── bootstrap.min.js
└── fonts/
    ├── glyphicons-halflings-regular.eot
    ├── glyphicons-halflings-regular.svg
    ├── glyphicons-halflings-regular.ttf
    ├── glyphicons-halflings-regular.woff
    └── glyphicons-halflings-regular.woff2
```

① 已编译的 CSS 和 JS
(bootstrap.*)

② 已编译压缩的 CSS 和 JS
(bootstrap.min.*)

CSS 源码映射表

(bootstrap.*.map)

③ Glyphicons 的图标字体，
在可选的主题中使用。

基本的html模板

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8"/>
  <meta name="viewport" content="width=device-width, initial-scale=1"/>
  <title>Bootstrap基本结构</title>
  <link href="css/bootstrap.min.css" rel="stylesheet"/>
  <script src="js/jquery.js"/> </script>
  <script src="js/bootstrap.min.js"/> </script>
</head>
<body>
  <h1>你好，世界！ </h1>
</body>
</html>
```


Bootstrap基本流程

使用Bootstrap基本流程：

下载

- Bootstrap文件



创建

- Html5文档结构



引入

- Bootstrap文件
- jQuery文件



Bootstrap浏览器/设备支持

设备支持：

	Chrome	Firefox	IE	Opera	Safari
Android	YES	YES	不适用	NO	不适用
iOS	YES	不适用	不适用	NO	YES
Mac OS X	YES	YES	不适用	YES	YES
Windows	YES	YES	YES*	YES	NO

03

Bootstrap CSS概览

HTML 5 文档类型 (Doctype)

全局设置：

1、Bootstrap 使用了一些 HTML5 元素和 CSS 属性。
必须使用HTML5文档类型。

```
<!DOCTYPE html>  
<html>  
...  
</html>
```

移动设备优先

2、为了确保适当的绘制和触屏缩放，需要在 `<head>` 之中添加 `viewport` 元数据标签。

- 设备上的 `viewport` 指设备的屏幕上能用来显示网页的那一块区域。
- 添加 `viewport` 元数据标签可确保适当的绘制和触屏缩放。

```
<meta name="viewport" content="width=device-width,  
initial-scale=1"/>
```

Viewport

Viewport 的属性如下：

`width=device-width`：表示宽度是设备屏幕的宽度

`initial-scale=1.0`：表示初始的缩放比例

`minimum-scale=0.5`：表示最小的缩放比例

`maximum-scale=2.0`：表示最大的缩放比例

`user-scalable=yes/no`：表示用户是否可以调整缩放比例

排版与链接

3、排版和链接

Bootstrap 排版、链接样式设置了基本的全局样式。分别是：

- 为 body 元素设置 background-color: #fff;
- 使用 @font-family-base、@font-size-base 和 @line-height-base 变量作为排版的基本参数
- 为所有链接设置了基本颜色 @link-color，并且当链接处于 :hover 状态时才添加下划线

scaffolding.less

布局容器

4、container 类、container-fluid类用于包裹页面上的内容。

.container 类用于固定宽度并支持响应式布局的容器。

.container-fluid 类用于 100% 宽度，占据全部视窗的容器。

```
@media (min-width: 1200px)
.container {
    width: 1170px;
}
@media (min-width: 992px)
.container {
    width: 970px;
}.....
```

```
.container {
    padding-right: 15px;
    padding-left: 15px;
    margin-right: auto;
    margin-left: auto;
}
```

!!两种容器类不能互相嵌套 demo21-2.html

04

栅格系统



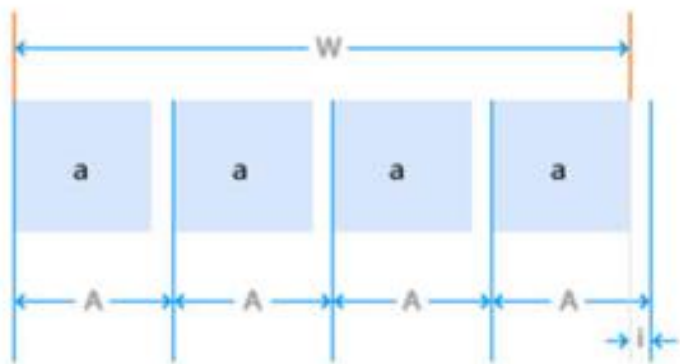
网格/栅格

- **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）。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**，栅格系统以规则的网格阵列来指导和规范网页中的版面布局以及信息分布。可以让网页的信息呈现更加美观易读，更具灵活、规范和可用性。

栅格系统的设计原理

在网页设计中，把宽度为“W”的页面分割成n个网格单元“a”，每个单元与单元之间的间隙设为“i”，把“a+i”定义“A”。

栅格化的设计原理



解：

$$W = (a \times n) + (n-1)i$$

由于 $a+i=A$
可得： $(A \times n) - i = W$

$$(A \times n) - i = W$$

A: 一个栅格单元的宽度

a: 一个栅格的宽度

$A=a+i$

n: 正整数

i: 栅格与栅格之间的间隙

W: 页面/区块的宽度

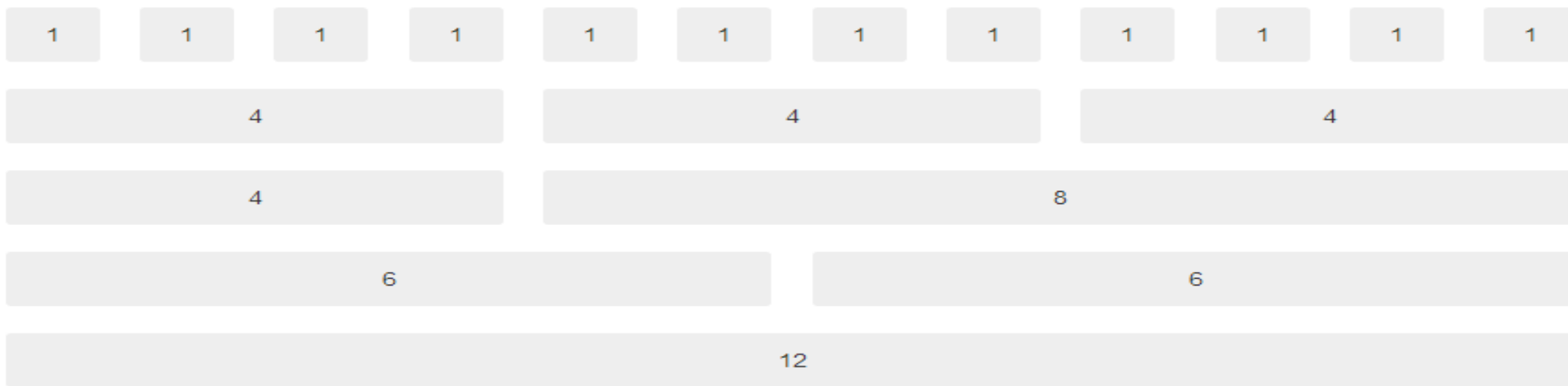
$$W = (a \times n) + (n-1)i$$

由于 $a+i=A$,

$$\text{可得: } (A \times n) - i = W$$

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，随着屏幕或视窗（viewport）尺寸的增加，系统会自动分为最多12列。



栅格系统


基本栅格： .row .col-md-*

```
<div class="container">  
  <div class="row">  
    <div class="col-md-4">...</div>  
    <div class="col-md-8">...</div>  
  </div>  
</div>
```



预定义类

demo21-3.html



栅格系统

注意：

- 1、行 (row) 必须要包含在容器 (container) 之内。
- 2、使用行 (row) 在水平方向创建一组列 (col) 。
- 3、内容应当放置于列内，而且，只有列可以作为行的直接子元素。
- 4、类似.row 和.col-md-4 这些预定义的栅格类可用来快速创建栅格布局。
- 5、通过设置 padding 创建列之间的间隔。
- 6、栅格系统中的列是通过指定1到12的值来表示其跨越的范围。例如，三个等宽的列可以使用三个.col-md-4来创建。
- 7、如果一“行 (row)”中包含了的“列 (col)”大于 12，多余的列所在的元素将被作为一个整体另起一行排列。

深入理解栅格系统

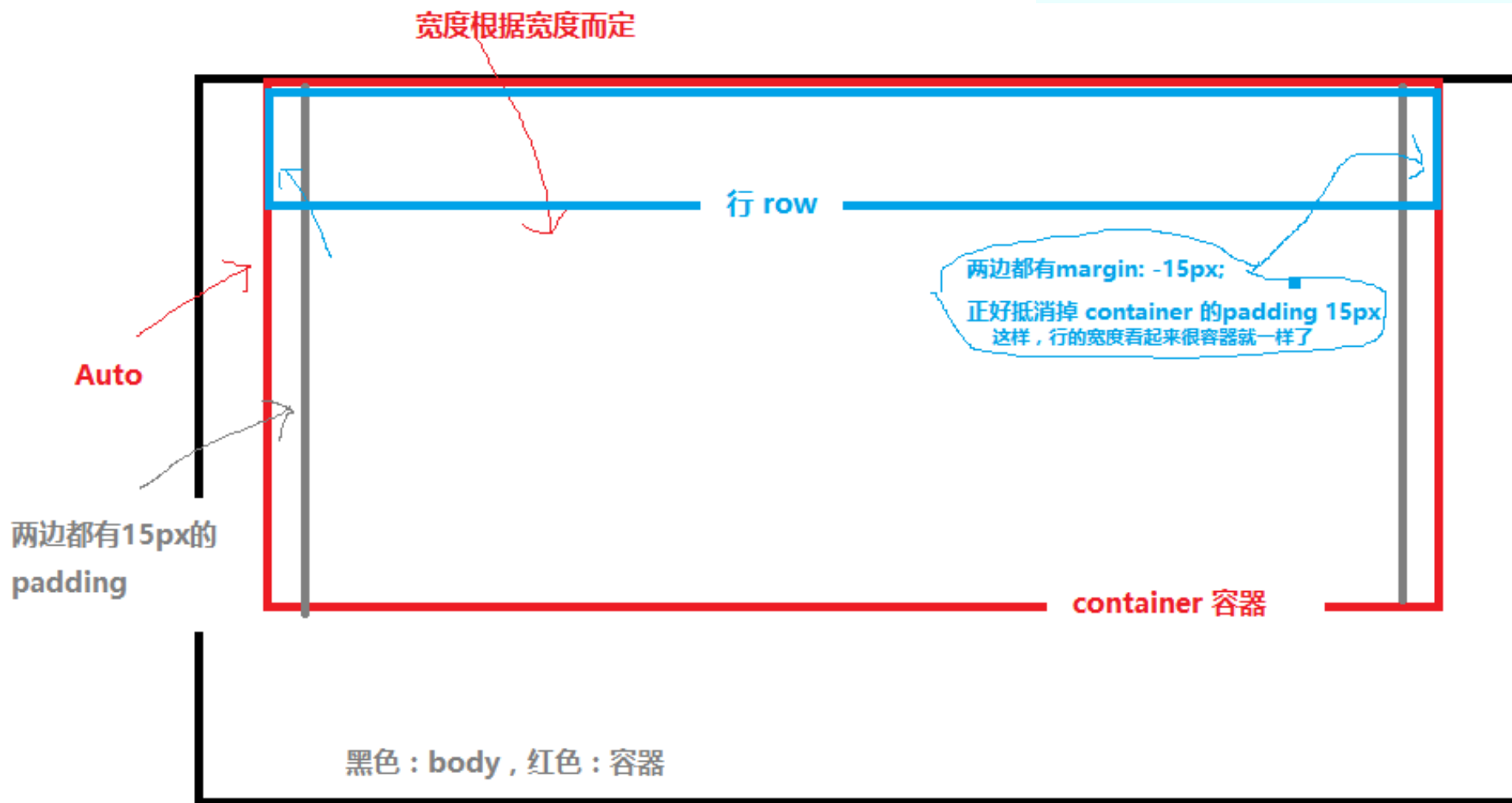
容器container 的设计如下：



深入理解栅格系统

行row 的设计如下：

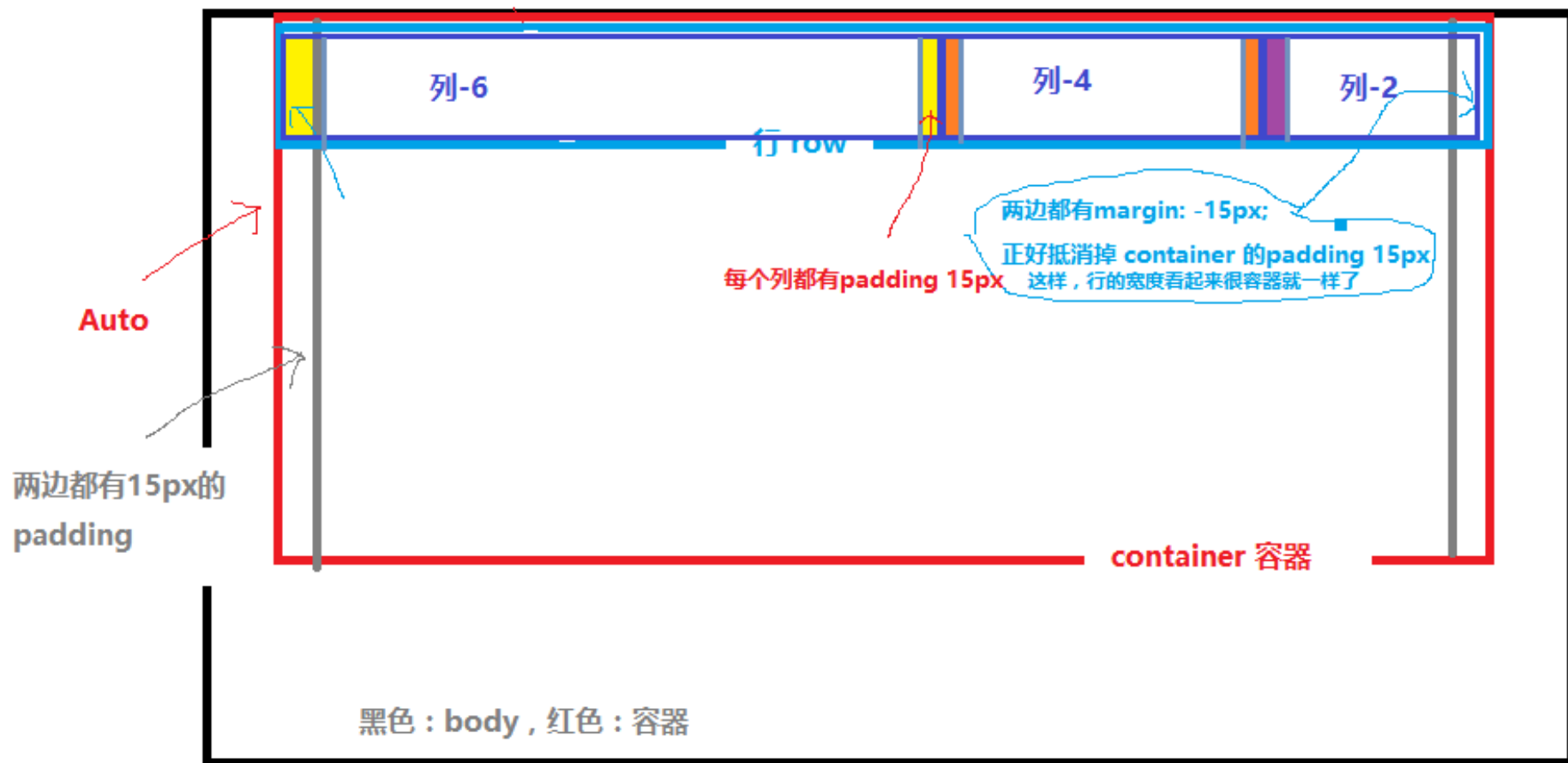
```
.row {  
    margin-right: -15px;  
    margin-left: -15px;  
}
```



深入理解栅格系统

列 Column的设计如下：

行被分为12份，你可以设定任意数字来调整



每列两边都有padding 15px



深入理解栅格系统

代码实现：col-xs-*：

```
.col-xs-1, .col-sm-1, .col-md-1, .col-lg-1, .col-xs-2, {  
/*后面代码省略，选择了所有的列*/  
position: relative;  
min-height: 1px;  
padding-right: 15px;  
padding-left: 15px; }  
.col-xs-1, .col-xs-2, .col-xs-3, .col-xs-4, .col-xs-5, .col-xs-6, .col-xs-7, .col-xs-8, .col-xs-9, .col-xs-10, .col-xs-11, .col-xs-12 {  
float: left; }
```

```
.col-xs-12 { width: 100%; }  
.col-xs-11 { width: 91.66666667%; }  
.col-xs-10 { width: 83.33333333%; }  
.col-xs-9 { width: 75%; }  
.col-xs-8 { width: 66.66666667%; }  
.col-xs-7 { width: 58.33333333%; }  
.col-xs-6 { width: 50%; }  
.col-xs-5 { width: 41.66666667%; }  
.col-xs-4 { width: 33.33333333%; }  
.col-xs-3 { width: 25%; }  
.col-xs-2 { width: 16.66666667%; }  
.col-xs-1 { width: 8.33333333%; }  
}
```

栅格系统

响应式设计：

可以根据用户的屏幕尺寸，合理地为现有及将来的各种设备提供最佳的浏览体验。

类型	布局宽度
大屏幕	大于等于1200px
默认	大于等于980px
平板	大于等于768px
手机到平板	小于等于767px
手机	小于等于480px



栅格选项

	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C		
容器最大宽度	None (auto)	750px	970px	1170px
类前缀	.col-xs-	.col-sm-	.col-md-	.col-lg-
列数	12			
最大列宽	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			

可同时使用

栅格选项



demo21-4.html

栅格选项

如果在一个 .row 内包含的列 (column) 大于12个，包含多余列的元素将作为一个整体单元被另起一行排列。

.col-xs-12 .col-md-8								.col-xs-6 .col-md-4			
.col-xs-6 .col-md-4				.col-xs-6 .col-md-4				.col-xs-6 .col-md-4			

.col-xs-12 .col-md-8											
.col-xs-6 .col-md-4											
.col-xs-6 .col-md-4						.col-xs-6 .col-md-4					
.col-xs-6 .col-md-4											

demo21-5.html

栅格系统

偏移列： `.col-md(xs/sm/lg)-offset-*`

将列向右侧偏移。本质是为当前元素增加了左侧的边距（margin）
例如，`.col-md-offset-4` 类将 `.col-md-4` 元素向右侧偏移了4个列的宽度。

```
<div class="row">  
  <div class="col-md-4">...</div>  
  <div class="col-md-3 col-md-offset-4">...</div>  
</div>
```

栅格系统

列嵌套： 在已有的.col-md- *内添加一个新的 .row并加入 .col-md- *

Container不能嵌套

嵌套列所包含的列加起来不能超过12列

Level 1: .col-md-9	
Level 2: .col-md-6	Level 2: .col-md-6

栅格系统

列排序： 通过使用`.col-md-push-*` `.col-md-pull-*`可以改变列的顺序。利用相对定位实现的，push是放在后面，pull是放在前面。

`.col-md-3 .col-md-pull-9`

`.col-md-9 .col-md-push-3`

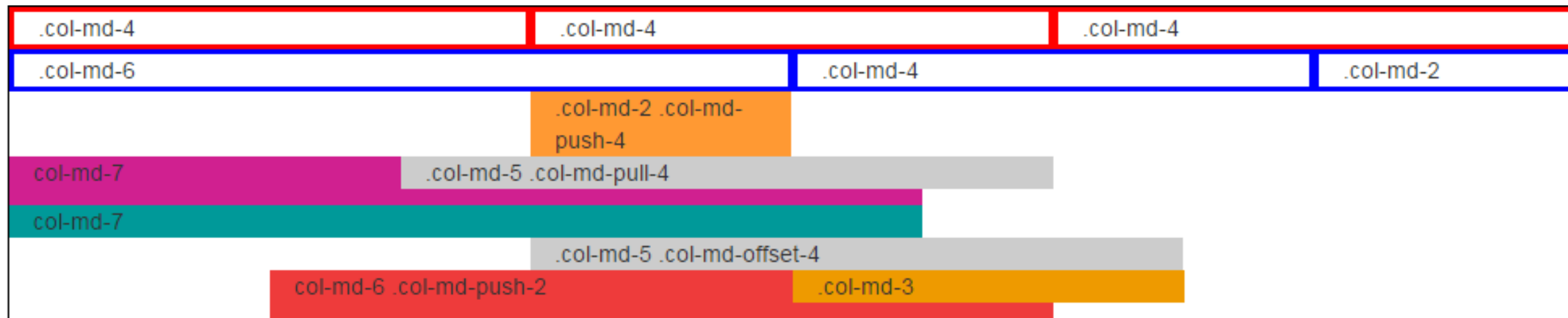
```
<div class="row">
```

```
<div class="col-md-9 col-md-push-3">.col-md-9 .col-md-push-3</div>
```

```
<div class="col-md-3 col-md-pull-9">.col-md-3 .col-md-pull-9</div>
```

```
</div>
```

栅格系统



栅格系统

禁用响应式设计步骤：

- 移除（或者不要添加）viewport <meta>。
- 通过为 .container 设置一个width值从而覆盖框架的默认width设置，例如width: 970px 。
- 如果使用了导航条，需要移除所有导航条的折叠和展开行为。
- 对于栅格布局，额外增加.col-xs-* 或替换掉.col-md-*和.col-lg-*。

练习

Content area1 The First

Bootstrap has a few easy ways to quickly get started, each one appealing to a different skill level and use case. Read through to see what suits your particular needs.

Content area2 The Second

Bootstrap has a few easy ways to quickly get started, each one appealing to a different skill level and use case. Read through to see what suits your particular needs.

Content area3 The Third

Bootstrap has a few easy ways to quickly get started, each one appealing to a different skill level and use case. Read through to see what suits your particular needs.

Report Table Form Detail

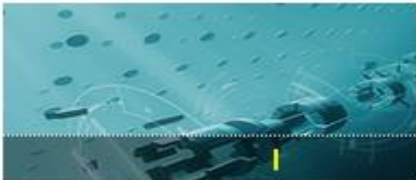
#	first	second	third	forth	fifth
1	penatibus et magnis	what	messages	56	on
2	penatibus et	you	information	44	down
3	nulla non	test	testing	32	up



Cum sociis natoque penatibus et magnis dis parturient montes

nascetur ridiculus mus. Nullam id dolor id nibh ultricies vehicula. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus.

Maecenas sed diam eget risus
Maecenas sed diam eget risus
Maecenas sed diam eget risus
Maecenas sed diam eget risus



THANKYOU

