

```

78 // ... trim(preg_replace('/\\\\\\\\/', '/', $image_src), '/');
79
80 $_SESSION['_CAPTCHA']['config'] = serialize($captcha_config);
81
82 return array(
83     'code' => $captcha_config['code'],
84     'image_src' => $image_src
85 );
86
87
88 if( !function_exists('hex2rgb') ) {
89     function hex2rgb($hex_str, $return_string = false, $separator = ',') {
90         $hex_str = preg_replace("/[^0-9A-Fa-f]/", '', $hex_str); // Gets a proper hex string
91         $rgb_array = array();
92         if( strlen($hex_str) == 6 ) {
93             $color_val = hexdec($hex_str);
94             $rgb_array['r'] = 0xFF & ($color_val >> 0x10);
95             $rgb_array['g'] = 0xFF & ($color_val >> 0x8);
96             $rgb_array['b'] = 0xFF & $color_val;
97         } elseif( strlen($hex_str) == 3 ) {
98             $rgb_array['r'] = hexdec(str_repeat(substr($hex_str, 0, 1), 2));
99             $rgb_array['g'] = hexdec(str_repeat(substr($hex_str, 1, 1), 2));
100             $rgb_array['b'] = hexdec(str_repeat(substr($hex_str, 2, 1), 2));
101         } else {
102             return false;
103         }
104         return $return_string ? implode($separator, $rgb_array) : $rgb_array;
105     }
106 }
107
108 // Draw the image
109 if( isset($_GET['captcha']) ) {
110     // ...
111 }

```

CREATIVE INTERACTION DESIGN

TANG, Xueting

Today's Class Is all About Hands-on Exercises





Continuing with Bootstrap

Getting to know Bootstrap

B

Home Documentation Examples Icons Themes Expo Blog

v4.4



Download

Bootstrap


Build responsive, mobile-first projects on the web with the world's most popular front-end component library.

Bootstrap is an open source toolkit for developing with HTML, CSS, and JS. Quickly prototype your ideas or build your entire app with our Sass variables and mixins, responsive grid system, extensive prebuilt components, and powerful plugins built on jQuery.

Get started

Download

Currently v4.4.1



- 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
- 最新版本为Bootstrap4，仅支持 IE9 以上以及 iOS7以上版本的浏览器

Getting to know Bootstrap

History

- Originally created by a designer and a developer at Twitter.
- Bootstrap was created at Twitter in mid-2010 by [@mdo](#) and [@fat](#)
- Now Bootstrap is maintained by a [small team of developers](#) on GitHub.

重要点

```
<!doctype html>
<html lang="en">
...
</html>
```

- 要求使用HTML5文档类型
- 移动设备优先，需在HTML的head之中添加viewport meta标签
`<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">`
- 容器 (.container)
- 栅格系统 (Grid system)

Getting started

Content Delivery Network 内容分发网络

1. 通过引用BootstrapCDN快速开始

Use the lasted version from official website
使用Bootstrap中文网构建的CND加速服务

2. 下载「用于生产环境的Bootstrap」到本地

Official Bootstrap website (v.4.4.1)
Bootstrap中文网 (v.3.3.7)

3. 其他方式下载

通过Bower进行安装

通过npm进行安装

通过Composer进行安装



偶然发现国内一个团队专门翻译了最新
Bootstrap4的中文使用手册，可参考

<https://code.z01.com/v4/>

开发者最好向前看

Starter Template

```
<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/
bootstrap.min.css" integrity="sha384-gg0yR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/
iJTQU0hcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">

    <title>My first Bootstrap Website</title>
  </head>
  <body>
    <h1>Hello, world!</h1>

    <!-- Optional JavaScript -->
    <!-- jQuery first, then Popper.js, then Bootstrap JS -->
    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/
X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></
script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9W01cLHTMGA3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJGzIxFDs4x0xIM+B07jRM"
crossorigin="anonymous"></script>
  </body>
</html>
```

Source: <https://code.z01.com/v4/docs/>

Part 1

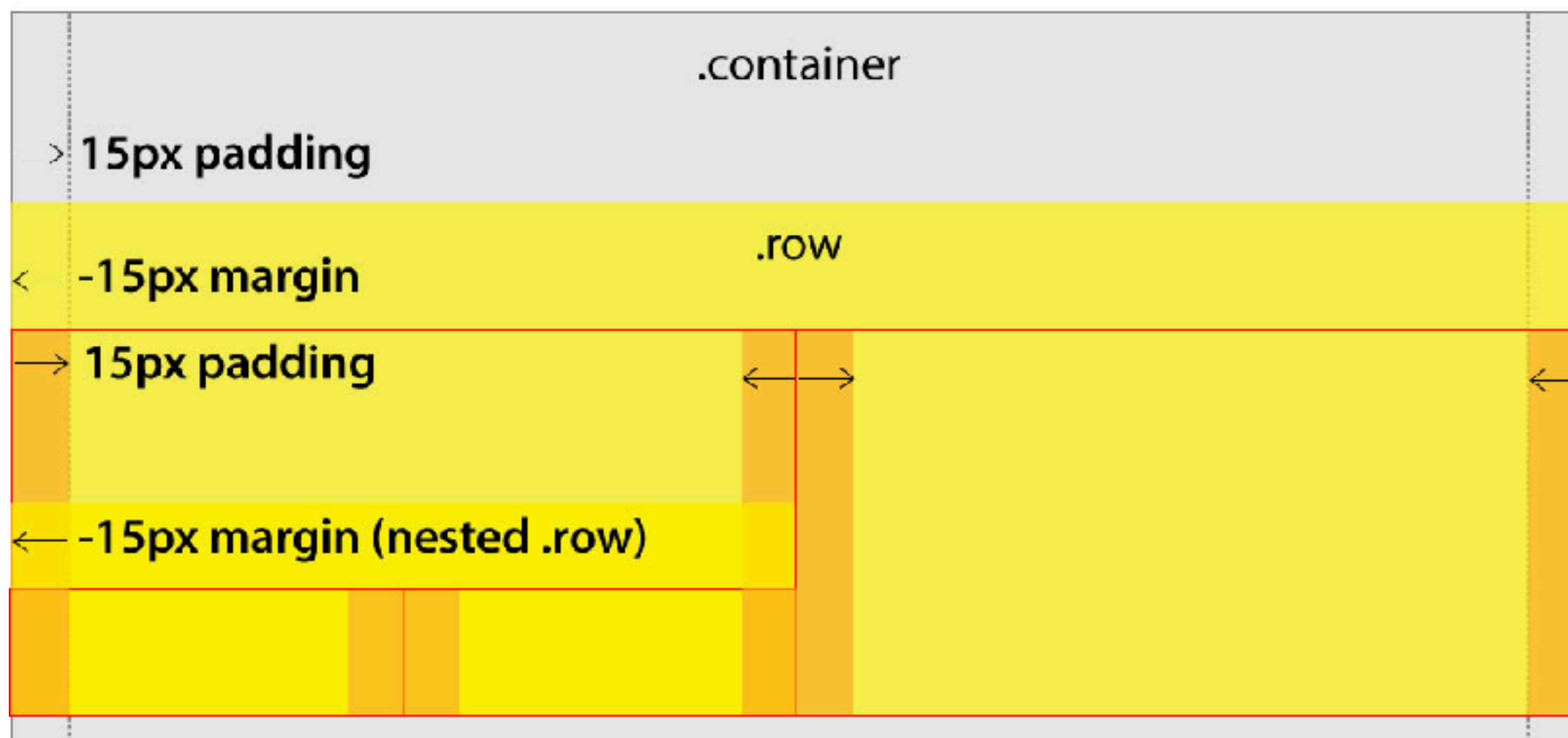
Layout 布局

- Container 容器
- Grid system 栅格/网格系统



Container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容器

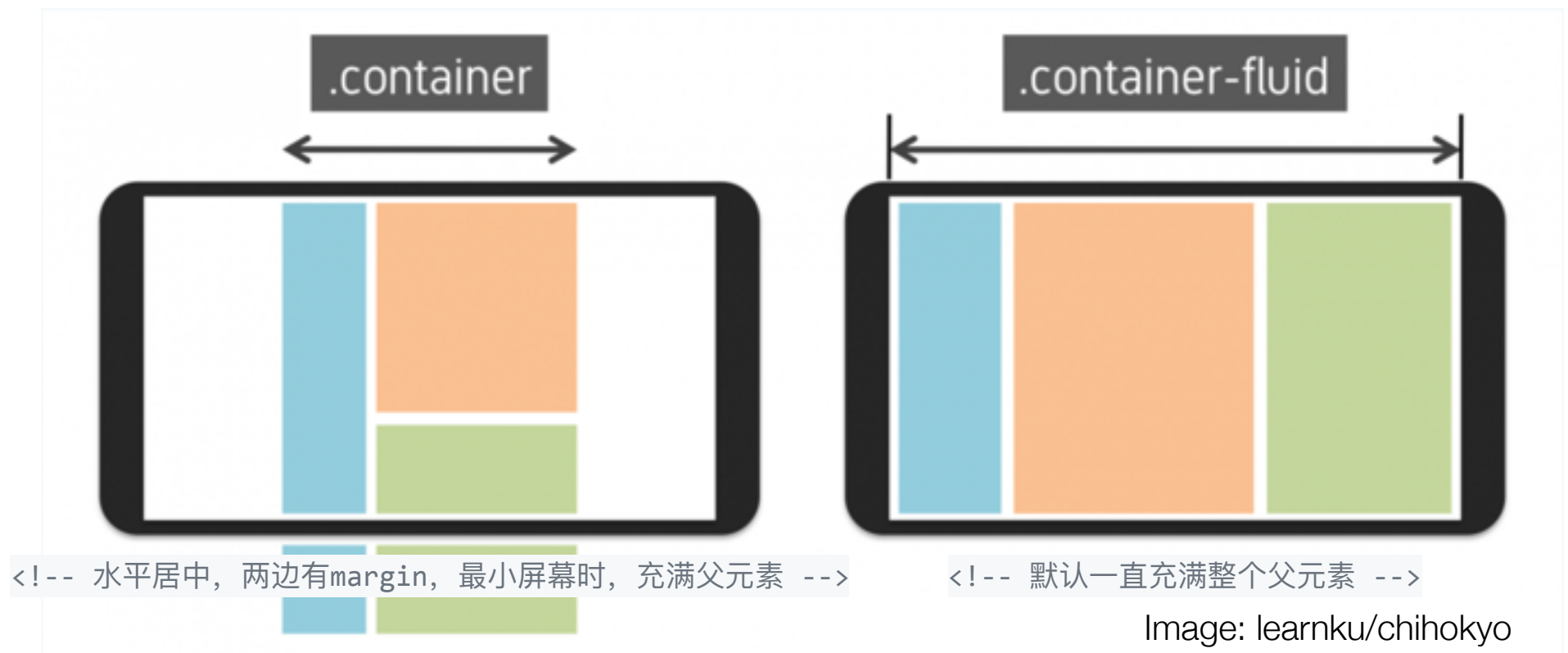
Bootstrap提供两类container:

- `.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

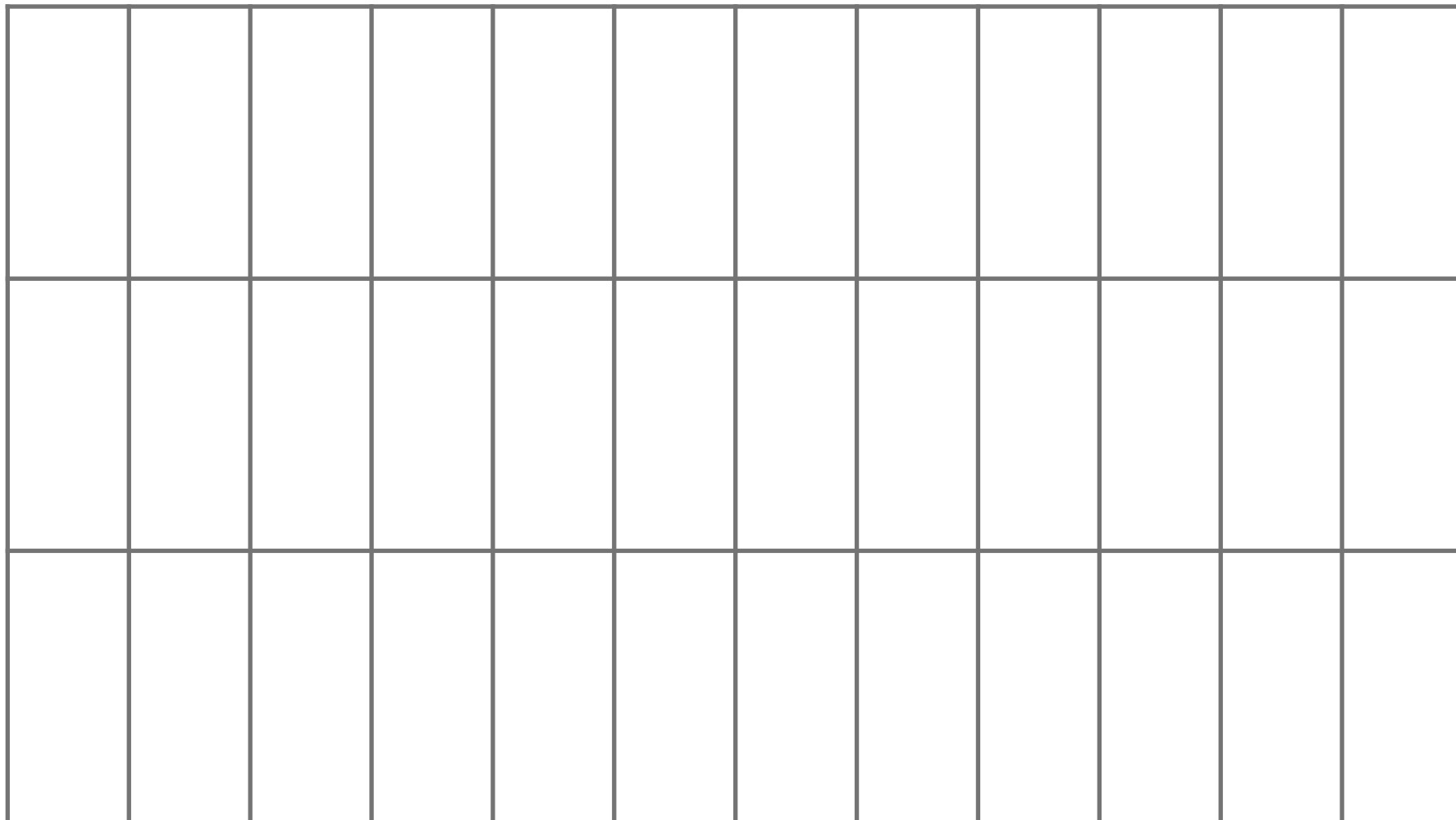
```
<div class="container-fluid">  
  ...  
</div>
```



Grid System 栅格/网格系统

- Grid system是Bootstrap**响应式**布局方案的核心，使用一系列

容器的行、列来布局和对齐内容；
- “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**。”

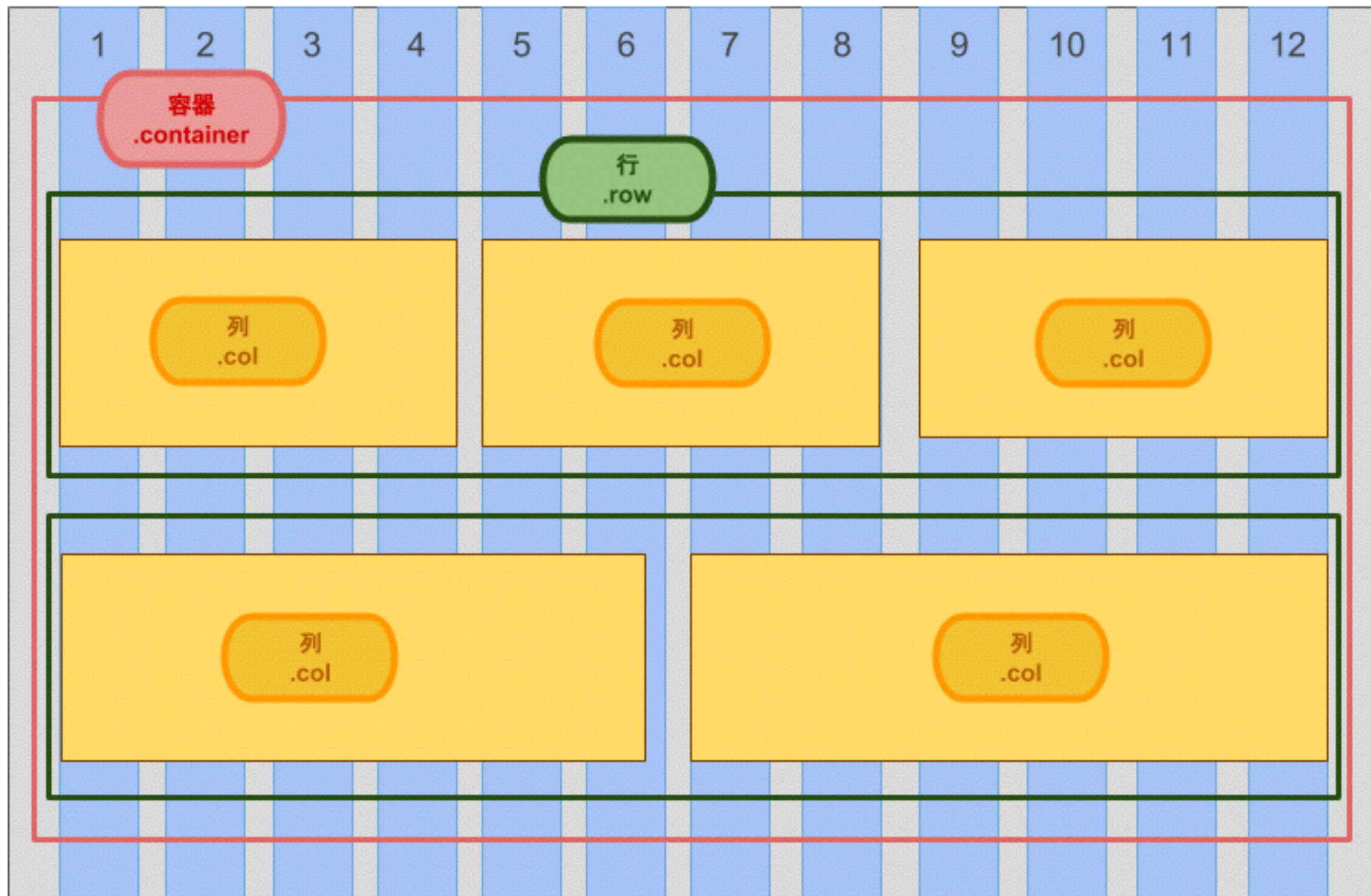


Grid System 栅格/网格系统

Responsive web design响应式网页设计



Grid System 栅格/网格系统



Grid system中的
三个重要“元素”

.container

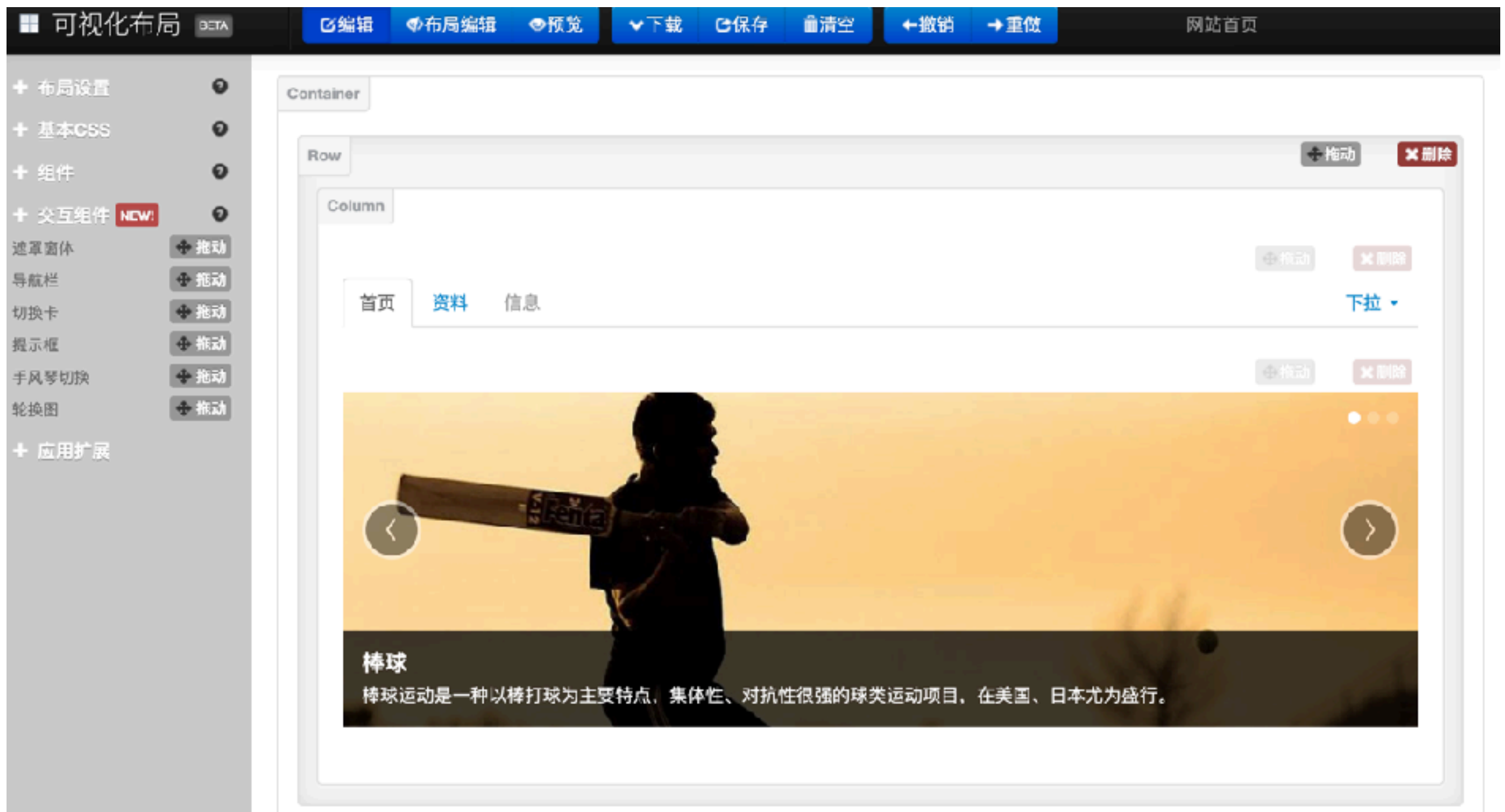
.row

.col

Grid System 栅格/网格系统

Have fun with Bootstrap可视化布局系统：

<http://www.bootcss.com/p/layoutit/>



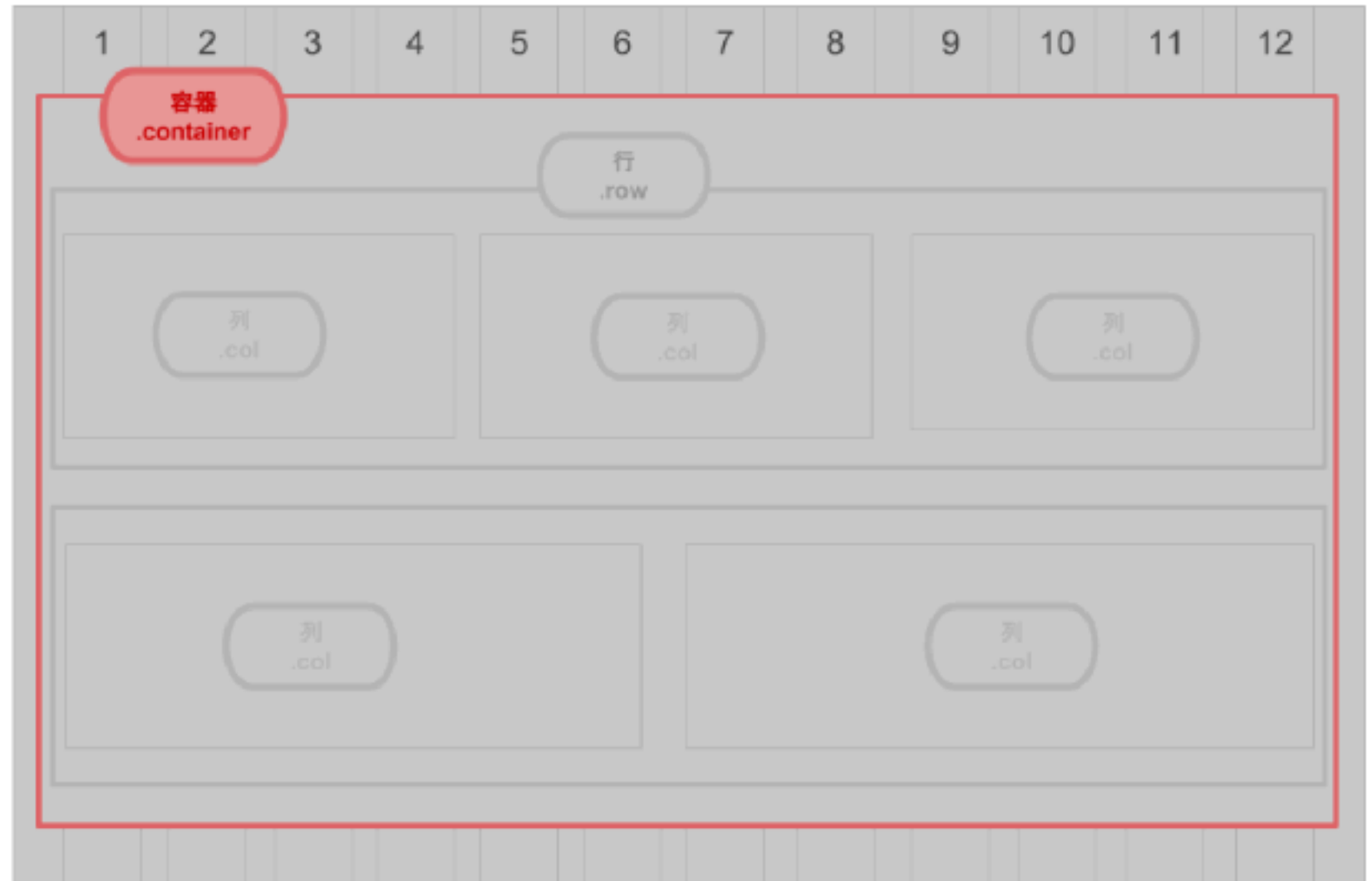
Grid System 栅格/网格系统

1

.container 容器

简单理解：

- 容器尺寸和浏览器页面尺寸不一样
- 容器尺寸小于浏览器页面



Grid System 栅格/网格系统

.container 容器

- Additional classes added in Bootstrap v4.4 allow containers that are 100% wide until **a particular breakpoint** (分界点阈值) .
- See them in action and compare them in [Grid example](#).

	超小屏幕 <576px	小屏幕 ≥576px	中等屏幕 ≥768px	大屏幕 ≥992px	超大屏幕 ≥1200px
<code>.container</code>	100%	540px	720px	960px	1140px
<code>.container-sm</code>	100%	540px	720px	960px	1140px
<code>.container-md</code>	100%	100%	720px	960px	1140px
<code>.container-lg</code>	100%	100%	100%	960px	1140px
<code>.container-xl</code>	100%	100%	100%	100%	1140px
<code>.container-fluid</code>	100%	100%	100%	100%	100%

Grid System 栅格/网格系统

About **responsive breakpoints**

- Bootstrap uses a handful of [media queries](#) to create sensible breakpoints for layouts and interfaces. These breakpoints are mostly based on minimum viewport widths and allow us to scale up elements as the viewport changes.
- Bootstrap primarily uses the following media query ranges—or breakpoints—in our source Sass files for our layout, grid system, and components.

```
// Extra small devices (portrait phones, less than 576px)
// No media query for `xs` since this is the default in Bootstrap

// Small devices (landscape phones, 576px and up)
@media (min-width: 576px) { ... }

// Medium devices (tablets, 768px and up)
@media (min-width: 768px) { ... }

// Large devices (desktops, 992px and up)
@media (min-width: 992px) { ... }

// Extra large devices (large desktops, 1200px and up)
@media (min-width: 1200px) { ... }
```


Grid System 栅格/网格系统

About **media queries** (媒体查询)

- Bootstrap栅格系统基于media queries实现，即根据当前设备以及尺寸等信息控制当前HTML采用哪种 css样式
- Media query语法

@media media type and|not|only (media feature) { CSS-Code; }

all
print
screen

链接media type与
media feature的逻
辑值

device-aspect-ratio
device-height
height
resolution
max-
...

- 对于media query内容若想详细了解，可查看以下几种简易解释

MDN: https://developer.mozilla.org/en-US/docs/Web/CSS/Media_Queries/Using_media_queries

CSDN: https://blog.csdn.net/qq_19865749/article/details/78305763

简书: <https://www.jianshu.com/p/15ac42fd959f>

Grid System 栅格/网格系统

About **media queries** (媒体查询)

- Try it & have fun:

```
body {  
  background-color: grey;  
}
```

```
@media screen and (max-width: 960px) { body {  
  background-color: green;  
} }
```

```
@media screen and (max-width: 768px) { body {  
  background-color: orange;  
} }
```

```
@media screen and (max-width: 550px) { body {  
  background-color: yellow;  
} }
```

Grid System 栅格/网格系统

2 .row行

最简单，上下排列

要点：

.row要在.container
里面

.col要在.row里面

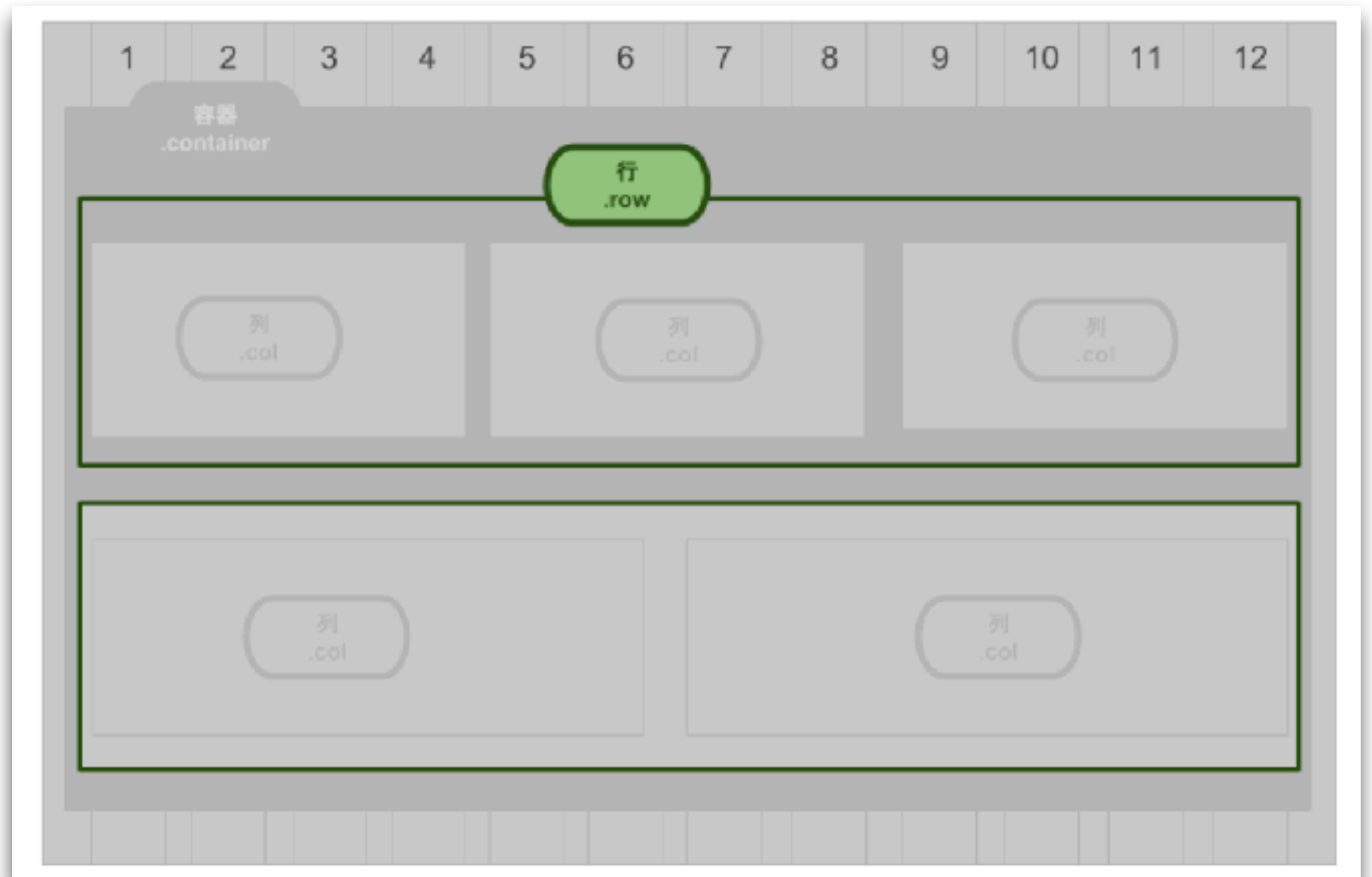


Image: laravel-china.org

Grid System 栅格/网格系统

3

.col列

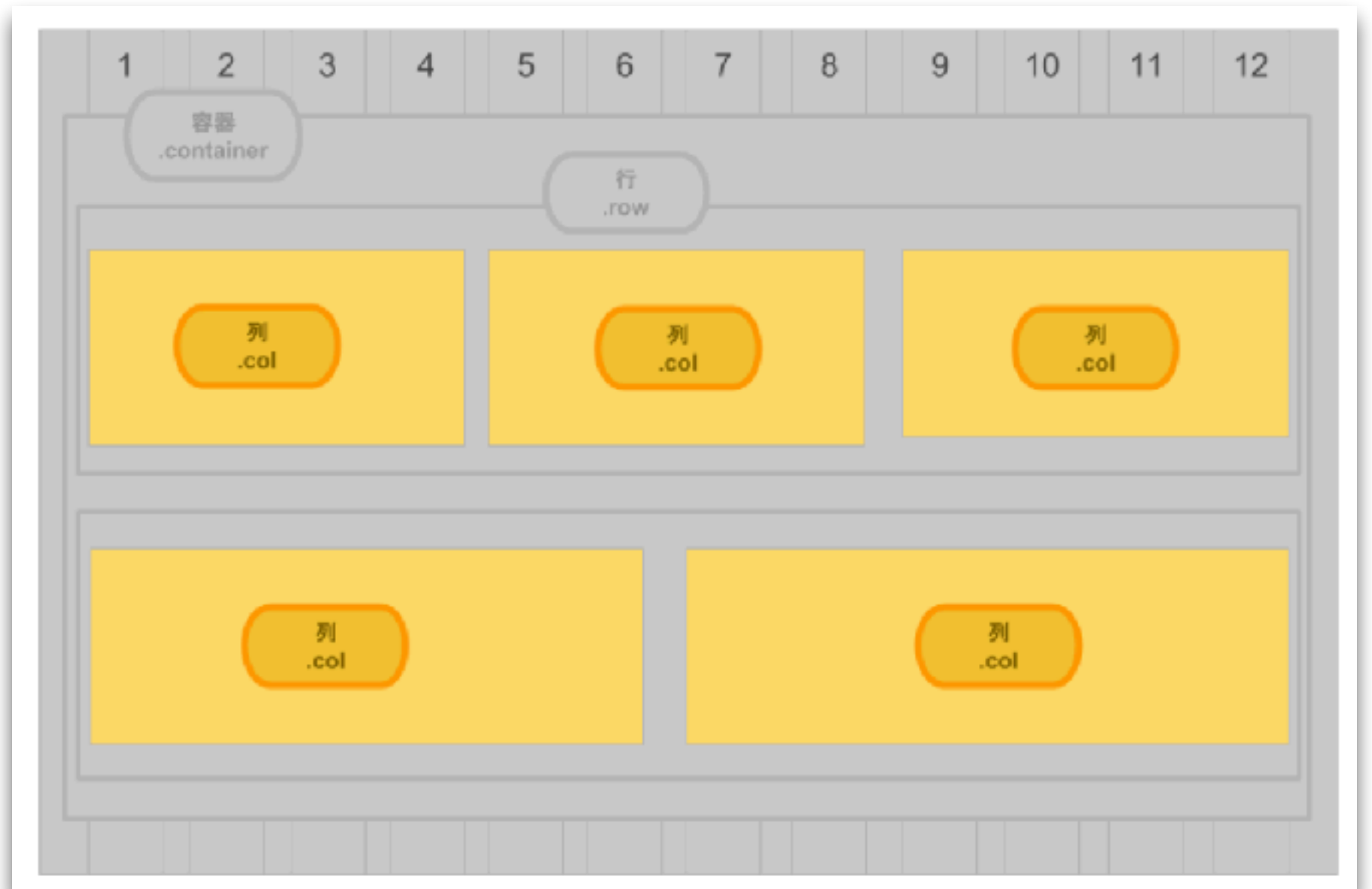
- 最繁琐，入门阶段理解原理可初步使用就好

• 写法：

col-容器类-栅格数

• 例子：

```
<div class="col-sm-6">  
</div>
```



Grid System 栅格/网格系统

Grid Options 栅格参数

See how aspects of the Bootstrap grid system work across multiple devices with a handy table.

	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px
Max container width	None (auto)	540px	720px	960px	1140px
Class prefix	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-
# of columns	12				
Gutter width	30px (15px on each side of a column)				
Nestable	Yes				
Column ordering	Yes				

	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			
.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			

Bootstrap英文官网

Bootstrap中文网

Grid System 栅格/网格系统

How does Bootstrap's Grid system works?

- Containers provide a means to center and horizontally pad your site's contents. Use `.container` for a responsive pixel width or `.container-fluid` for `width: 100%` across all viewport and device sizes.
- Rows are wrappers for columns. Each column has horizontal `padding` (called a gutter) for controlling the space between them. This `padding` is then counteracted on the rows with negative margins. This way, all the content in your columns is visually aligned down the left side.
- In a grid layout, content must be placed within columns and only columns may be immediate children of rows.
- Thanks to flexbox, grid columns without a specified `width` will automatically layout as equal width columns. For example, four instances of `.col-sm` will each automatically be 25% wide from the small breakpoint and up. See the [auto-layout columns](#) section for more examples.
- Column classes indicate the number of columns you'd like to use out of the possible 12 per row. So, if you want three equal-width columns across, you can use `.col-4`.
- Column `widths` are set in percentages, so they're always fluid and sized relative to their parent element.
- Columns have horizontal `padding` to create the gutters between individual columns, however, you can remove the `margin` from rows and `padding` from columns with `.no-gutters` on the `.row`.
- To make the grid responsive, there are five grid breakpoints, one for each [responsive breakpoint](#): all breakpoints (extra small), small, medium, large, and extra large.
- Grid breakpoints are based on minimum width media queries, meaning **they apply to that one breakpoint and all those above it** (e.g., `.col-sm-4` applies to small, medium, large, and extra large devices, but not the first `xs` breakpoint).
- You can use predefined grid classes (like `.col-4`) or [Sass mixins](#) for more semantic markup.

Grid System 栅格/网格系统

Bootstrap栅格系统工作原理

- “行 (row)”必须包含在 `.container` (固定宽度) 或 `.container-fluid` (100% 宽度) 中, 以便为其赋予合适的排列 (alignment) 和内补 (padding)。
- 通过“行 (row)”在水平方向创建一组“列 (column)”。
- 你的内容应当放置于“列 (column)”内, 并且, 只有“列 (column)”可以作为行 (row)”的直接子元素。
- 类似 `.row` 和 `.col-xs-4` 这种预定义的类, 可以用来快速创建栅格布局。Bootstrap 源码中定义的 mixin 也可以用来创建语义化的布局。
- 通过为“列 (column)”设置 `padding` 属性, 从而创建列与列之间的间隔 (gutter)。通过为 `.row` 元素设置负值 `margin` 从而抵消掉为 `.container` 元素设置的 `padding`, 也就间接为“行 (row)”所包含的“列 (column)”抵消掉了 `padding`。
- 负值的 `margin`就是下面的示例为什么是向外突出的原因。在栅格列中的内容排成一行。
- 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-xs-4` 来创建。
- 如果一“行 (row)”中包含了的“列 (column)”大于 12, 多余的“列 (column)”所在的元素将被作为一个整体另起一行排列。
- 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。因此, 在元素上应用任何 `.col-md-*` 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。因此, 在元素上应用任何 `.col-lg-*` 不存在, 也影响大屏幕设备。

Grid System 栅格/网格系统

Bootstrap4基本网格结构

```
<!-- 第一个例子：控制列的宽度及在不同的设备上如何显示 -->
<div class="row">
  <div class="col-*-*"></div>
</div>
<div class="row">
  <div class="col-*-*"></div>
  <div class="col-*-*"></div>
  <div class="col-*-*"></div>
</div>

<!-- 第二个例子：或让 Bootstrap 者自动处理布局 -->
<div class="row">
  <div class="col"></div>
  <div class="col"></div>
  <div class="col"></div>
</div>
```


Grid System 栅格/网格系统

Now let's get started!

a. 创建相等宽度的列，Bootstrap 自动布局

```
<div class="container-fluid">
  <h1>Hello World!</h1>
  <p>创建三个相等宽度的列！尝试在 class="row" 的 div 中添加新的 class="col"
div, 会显示四个等宽的列。</p>
  <div class="row">
    <div class="col" style="background-color:lavender;">.col</div>
    <div class="col" style="background-color:orange;">.col</div>
    <div class="col" style="background-color:lavender;">.col</div>
  </div>
</div>
```

Grid System 栅格/网格系统

b. 等宽响应式列

<!-- 以下实例演示了如何在平板及更大屏幕上创建等宽度的响应式列。在移动设备上，即屏幕宽度小于 576px 时，四个列将会上下堆叠排版 -->

```
<div class="container-fluid">
  <h1>Hello World!</h1>
  <p>重置浏览器大小查效果。</p>
  <p> 在移动设备上，即屏幕宽度小于 576px 时，四个列将会上下堆叠排版。</p>
  <div class="row">
    <div class="col-sm-3" style="background-color:lavender;">.col-sm-3</div>
    <div class="col-sm-3" style="background-color:lavenderblush;">.col-sm-3</div>
    <div class="col-sm-3" style="background-color:lavender;">.col-sm-3</div>
    <div class="col-sm-3" style="background-color:lavenderblush;">.col-sm-3</div>
  </div>
</div>
```

Grid System 栅格/网格系统

c. 不等宽响应式列

`<!-- 以下实例演示了在平板及更大屏幕上创建不等宽度的响应式列。 在移动设备上，即屏幕宽度小于 576px 时，两个列将会上下堆叠排版： -->`

```
<div class="container-fluid">
  <h1>Hello World!</h1>
  <p>重置浏览器大小查效果。</p>
  <p>在移动设备上，即屏幕宽度小于 576px 时，两个列将会上下堆叠排版。</p>
  <div class="row">
    <div class="col-sm-4" style="background-color:lavender;">.col-sm-4</div>
    <div class="col-sm-8" style="background-color:lavenderblush;">.col-sm-8</div>
  </div>
</div>
```

Grid System 栅格/网格系统

d. 平板和桌面

<!-- 以下实例演示了在桌面设备的显示器上两个列的宽度各占 50%，如果在平板端则左边的宽度为 25%，右边的宽度为 75%，在移动手机等小型设备上会堆叠显示。 -->

```
<div class="container-fluid">
```

```
  <h1>平板与桌面的网格布局</h1>
```

```
  <p>以下实例演示了在桌面设备的显示器上两个列的宽度各占 50%，如果在平板端则左边的宽度为 25%，右边的宽度为 75%，在移动手机等小型设备上会堆叠显示。
```

```
</p>
```

```
  <p>重置浏览器窗口大小，查看效果。</p>
```

```
  <div class="container-fluid">
```

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

```
      <div class="col-sm-3 col-md-6 bg-success">RUN00B</div>
```

```
      <div class="col-sm-9 col-md-6 bg-warning">菜鸟教程</div>
```

```
    </div>
```

```
  </div>
```

```
</div>
```

Grid System 栅格/网格系统

e. 平板、桌面、大桌面显示器、超大桌面显示器

<!-- 以下实例在平板、桌面、大桌面显示器、超大桌面显示器的宽度比例为分别为：25%/75%、50%/50%、33.33%/66.67%、16.67/83.33%，在移动手机等小型设备上会堆叠显示。 -->

```
<div class="container-fluid">
  <h1>平板、桌面、大桌面显示器、超大桌面显示器</h1>
  <p>以下实例在平板、桌面、大桌面显示器、超大桌面显示器的宽度比例为分别为：25%/75%、50%/50%、33.33%/66.67%、16.67/83.33%，在移动手机等小型设备上会堆叠显示。
</p>
  <p>重置浏览器窗口大小，查看效果。</p>
  <div class="container-fluid">
    <div class="row">
      <div class="col-sm-3 col-md-6 col-lg-4 col-xl-2 bg-success">RUN00B</div>
      <div class="col-sm-9 col-md-6 col-lg-8 col-xl-10 bg-warning">菜鸟教程</div>
    </div>
  </div>
</div>
```


Grid System 栅格/网格系统

f. 手机、平板、桌面（如何不让小屏幕上的列堆叠在一起）

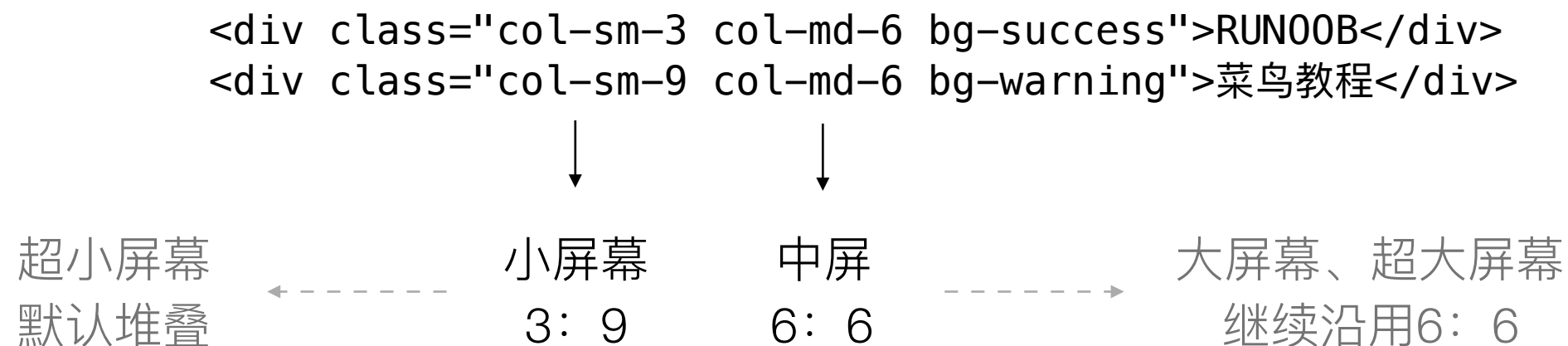
<!-- 是否不希望在小屏幕设备上所有列都堆叠在一起？那就使用针对超小屏幕、小屏幕和中等屏幕设备所定义的类，来创建更加动态和强大的布局吧。-->

```
<div class="container">
  <div class="row">
    <div class="col-xs-12 col-sm-6 col-md-8 bg-success">.col-xs-12 .col-
sm-6 .col-md-8</div>
    <div class="col-xs-6 col-sm-12 col-md-4 bg-warning">.col-xs-6 .col-
sm-12 .col-md-4</div>
  </div>
  <div class="row">
    <div class="col-xs-6 col-sm-4 bg-success">.col-xs-6 .col-sm-4</div>
    <div class="col-xs-6 col-sm-4 bg-primary">.col-xs-6 .col-sm-4</div>
    <div class="col-xs-6 col-sm-4 bg-success">.col-xs-6 .col-sm-4</div>
  </div>
</div>
```

Grid System 栅格/网格系统

小结:

- 若想分别使得在手机、平板、桌面、大屏幕等显示屏中排列方式不同，则需分别定义`col-sm-*`，`col-md-*`，`col-lg-*`，`col-xl-*`的布局方式；
- 响应式列重置，在一个`.row`仍然存在列超过12时，`col-xs-6 * 3=18`。多余列的元素将作为一个*整体单元被另起一行排列；
- 若未定义超小屏和超大屏的布局方式，则其排列方式分别为堆叠和继续沿用中屏的布局方式。



Grid System 栅格/网格系统

- 关于Grid system还有不少内容：如offsetting列偏移、alignment对齐、nesting嵌套、ordering排序.....

- 理解原理后，这些效果将不难学习，如有兴趣可参考：

Bootstrap中文网 <https://v3.bootcss.com/css/#grid-offsetting>

Bootstrap官网 <https://getbootstrap.com/docs/4.4/layout/grid/#alignment>

Part 2

Content 内容

- Typography 排版
- Colors 颜色
- Tables 表格
- Images 图片



全局CSS样式

Bootstrap 排版、链接样式设置了基本的全局样式。分别是：

- 移除body的margin声明
- 设置body的背景色为白色 background-color: #fff;
- 为排版设置了基本的字体@font-family-base、字号@font-size-base与行高@line-height-base
- 设置全局链接颜色 @link-color，且当链接处于悬浮状态时: hover 才会显示下划线

Typography 排版

- Bootstrap 4 默认的 **font-size** 为 16px, **line-height** 为 1.5
- 默认的 **font-family** 为 "Helvetica Neue", Helvetica, Arial, Sans-serif
- 此外, 所有的 `<p>` 元素 `margin-top: 0`, `margin-bottom: 1rem` (16px)

- Bootstrap 4 定义了 h1-h6 的标题样式

`<h1>`h1 Bootstrap 标题 (2.5rem = 40px)`</h1>`

`<h2>`h2 Bootstrap 标题 (2rem = 32px)`</h2>`

`<h3>`h3 Bootstrap 标题 (1.75rem = 28px)`</h3>`

`<h4>`h4 Bootstrap 标题 (1.5rem = 24px)`</h4>`

`<h5>`h5 Bootstrap 标题 (1.25rem = 20px)`</h5>`

`<h6>`h6 Bootstrap 标题 (1rem = 16px)`</h6>`

- Display 标题类 (h1-h4) `.display-*`
- 自定义标题备注 `<small>`
- 高亮 `<mark>`
- 引用 `<blockquote>` 中添加 `.blockquote` 类
- 表格描述 `<dl>`

关于 rem 详情可阅读:

web app 变革之 rem - <https://www.jianshu.com/p/22d2d6fd3187>

Typography 排版

Bootstrap 4 中更多排版类实例

Class	Description
<code>.font-weight-bold</code>	加粗文本
<code>.font-weight-light</code>	更细文本
<code>.font-italic</code>	斜体文本
<code>.lead</code>	中心段落内容突出
<code>.text-left, .text-center, .text-right, .text-justify</code>	左对齐, 居中, 右对齐, 文本对齐
<code>.text-lowercase</code>	文本大写
<code>.text-uppercase</code>	文本小写
<code>.text-capitalize</code>	首字母大写

Colors 颜色

- Bootstrap 4 提供了一些有代表意义的颜色类: `.text-muted`, `.text-primary`, `.text-success`, `.text-info`, `.text-warning`, `.text-danger`, `.text-secondary`, `.text-white`, `.text-dark` 和 `.text-light`

```
<p class="text-primary">.text-primary</p>
<p class="text-secondary">.text-secondary</p>
<p class="text-success">.text-success</p>
<p class="text-danger">.text-danger</p>
<p class="text-warning">.text-warning</p>
<p class="text-info">.text-info</p>
<p class="text-dark">.text-dark</p>
<p class="text-body">.text-body</p>
<p class="text-muted">.text-muted</p>
<p class="text-black-50">.text-black-50</p>
<p class="text-white-50 bg-dark">.text-white-50</p>
```

- 可在链接中使用
- 可在背景颜色中使用 `.bg-primary`
- 背景渐变 `.bg-gradient-primary`

Tables 表格

- Bootstrap 4 基础表格 `.table`
- 带边框表格 `.table table-bordered`
- 深色背景表格 `.table table-dark`
- 表头背景色处理，在<thead>中添加`.thead-dark`或`.thead-light`类
- 鼠标悬停，为每一行增加灰色背景 `.table table-hover`
- 为深色背景表格添加鼠标悬停 `.table table-dark table-hover`
- 响应式表格 `.table-responsive`

Images 图片

- Bootstrap 4 为图片设置了一些简单好玩的形状：
- 椭圆图片 `.rounded-circle`
- 圆角图片 `.rounded`
- 缩略图 `.thumbnail`
- 图片右对齐 `.float-right` 图片左对齐 `.float-left`
- 响应式图片：

在 `` 标签中添加 `.img-fluid` 类来设置响应式图片

`.img-fluid` 类设置了 `max-width: 100%; height: auto;`

e.g. ``

Part 3

Components 组件

- Jumbotron 超大屏幕
- Navs, Navbar&Breadcrumb 导航系列
- Dropdown 下拉菜单
- Carousel 轮播效果
- Buttons & Button group 按钮系列
- Forms & Input group 表单&输入框组
- Card 卡片
- Media-object 媒体对象
- Collapse 折叠
- Alert 警告提示
- Modal 弹出模态框



Jumbotron 超大屏幕

- Bootstrap 4 提供的Jumbotron实现构建超大的Hero界面，醒目的在网站上展示关键的营销信息，轻量、灵活。
- 通过在 `<div>` 元素 中添加 `.jumbotron` 类来创建 jumbotron:

```
<div class="jumbotron">
  <h1 class="display-3">Hello, world!</h1>
  <p class="lead">This is a simple hero unit, a simple jumbotron-style component for
calling extra attention to featured content or information.</p>
  <hr class="my-4">
  <p>It uses utility classes for typography and spacing to space content out within the
larger container.</p>
  <p class="lead">
    <a class="btn btn-primary btn-lg" href="#" role="button">Learn more</a>
  </p>
</div>
```

- 想要让广告大屏幕占满当前显示浏览器全屏，只要添加 `.jumbotron-fluid` CSS 修饰符，并在里面添加一个 `.container` 或 `.container-fluid` 内容空间即可

```
<div class="jumbotron jumbotron-fluid">
  <div class="container">
    <h1 class="display-4">Fluid jumbotron</h1>
    <p class="lead">This is a modified jumbotron that occupies the entire horizontal space of
its parent.</p>
  </div>
</div>
```


Collapse 折叠

Bootstrap 4 折叠：通过按钮或锚点用作**触发器**，映射到切换的特定元素

`.collapse` 隐藏内容

`.collapsing` 带动态效果的切换

`.collapse.show` 显示内容

可以使用带 `href` 属性的连接或者带 `data-target` 属性的按钮来创建折叠效果，但这两种情况下 `data-toggle="collapse"` 属性都是必须的

```
<div class="container">
  <h2>简单的折叠</h2>
  <a href="#demo" class="btn btn-primary" data-toggle="collapse">简单的折叠</a>
  <div id="demo" class="collapse">
    Lorem ipsum dolor sit amet, consectetur adipisicing elit,
    sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,
    quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.
  </div>
</div>
```

- `.collapse` 类用于指定一个折叠元素 (实例中的 `<div>`); 点击按钮后会在隐藏与显示之间切换
- 控制内容的隐藏与显示，需要在 `<a>` 或 `<button>` 元素上添加 `data-toggle="collapse"` 属性。
`data-target="#id"` 属性是对应折叠的内容 (`<div id="demo">`)
- `<a>` 元素上你可以使用 `href` 属性来代替 `data-target` 属性

Navs, Navbar & Breadcrumb 导航系列

.nav 导航/滑动门

创建一个简单的水平导航栏：在 `` 元素上添加 `.nav` 类，在每个 `` 选项上添加 `.nav-item` 类，在每个链接上添加 `.nav-link` 类：

```
<ul class="nav">
  <li class="nav-item">
    <a class="nav-link active" href="#">Active</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Link</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Link</a>
  </li>
  <li class="nav-item">
    <a class="nav-link disabled" href="#">Disabled</a>
  </li>
</ul>
```

- 导航对齐： `.justify-content-center` 类导航居中 `.justify-content-end` 类导航右对齐
- 胶囊导航 `.nav-pills` 类可以将导航项设置成胶囊形状
- 选项卡使用 `.nav-tabs` 类可以将导航转化为选项卡

Navs, Navbar & Breadcrumb 导航系列

.navbar 导航栏

- 导航栏需要使用 .navbar 来定义，并使用 .navbar-expand{-sm|-md|-lg|-xl} 用于响应式布局以及使用 配色方案 Class
- 导航栏默认内容是流式的，使用 containers 容器来限制它们的水平宽度

<!-- 小屏幕上水平导航栏会切换为垂直的 -->

```
<nav class="navbar navbar-expand-sm bg-dark navbar-dark">
```

```
<!-- Links -->
```

```
<ul class="navbar-nav">
```

```
<li class="nav-item active">
```

```
<a class="nav-link" href="#">Active</a>
```

```
</li>
```

```
<li class="nav-item">
```

```
<a class="nav-link" href="#">Link 2</a>
```

```
</li>
```

```
<li class="nav-item">
```

```
<a class="nav-link" href="#">Link 3</a>
```

```
</li>
```

```
<li class="nav-item">
```

```
<a class="nav-link disabled" href="#">Disabled</a>
```

```
</li>
```

```
</ul>
```

```
</nav>
```

Navs, Navbar & Breadcrumb 导航系列

.navbar 导航栏

折叠导航栏：宽屏展开显示，小屏幕折叠导航栏，通过点击来显示导航选项

```
<nav class="navbar navbar-expand-md bg-dark navbar-dark">
  <!-- Brand -->
  <a class="navbar-brand" href="#">Navbar</a>

  <!-- Toggler/collapsible Button -->
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#collapsibleNavbar">
    <span class="navbar-toggler-icon"></span>
  </button>
  <!-- Navbar links -->
  <div class="collapse navbar-collapse" id="collapsibleNavbar">
    <ul class="navbar-nav">
      <li class="nav-item">
        <a class="nav-link active" href="#">Link</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Link</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Link</a>
      </li>
    </ul>
  </div>
</nav>
```

原理：

button-触发器

data-toggle-开关，什么事件触发

data-target-事件触发的对象

Navs, Navbar & Breadcrumb 导航系列

.navbar 导航栏

- 导航栏中设置下拉菜单
- 导航栏的表单与按钮
- 固定导航栏导航栏可以固定在头部或者底部

使用 **.fixed-top** 类来实现导航栏的固定：

```
<nav class="navbar navbar-expand-sm bg-dark navbar-dark fixed-top">
  ...
</nav>
```

.fixed-bottom 类用于设置导航栏固定在底部：

```
<nav class="navbar navbar-expand-sm bg-dark navbar-dark fixed-bottom">
  ...
</nav>
```

Navs, Navbar & Breadcrumb 导航系列

.breadcrumb 面包屑导航

面包屑导航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，是在用户界面中的一种导航辅助。

```
<ol class="breadcrumb">
  <li class="breadcrumb-item active">Home</li>
</ol>
<ol class="breadcrumb">
  <li class="breadcrumb-item"><a href="#">Home</a></li>
  <li class="breadcrumb-item active">Library</li>
</ol>
<ol class="breadcrumb">
  <li class="breadcrumb-item"><a href="#">Home</a></li>
  <li class="breadcrumb-item"><a href="#">Library</a></li>
  <li class="breadcrumb-item active">Data</li>
</ol>

<!-- 也可不用列表形式 -->
<nav class="breadcrumb">
  <a class="breadcrumb-item" href="#">Home</a>
  <a class="breadcrumb-item" href="#">Library</a>
  <a class="breadcrumb-item" href="#">Data</a>
  <span class="breadcrumb-item active">Bootstrap</span>
</nav>
```


Dropdown 下拉菜单

Bootstrap 4的下拉菜单依赖于 popper.min.js

```
<div class="dropdown">
  <button type="button" class="btn btn-primary dropdown-toggle" data-toggle="dropdown">
    Dropdown button
  </button>
  <div class="dropdown-menu">
    <a class="dropdown-item" href="#">Link 1</a>
    <a class="dropdown-item" href="#">Link 2</a>
    <a class="dropdown-item" href="#">Link 3</a>
  </div>
</div>
```

.dropdown 类用来指定一个下拉菜单。

我们可以使用一个按钮或链接来打开下拉菜单，按钮或链接需要添加 **.dropdown-toggle** 和 **data-toggle="dropdown"** 属性。

<div> 元素上添加 **.dropdown-menu** 类来设置实际下拉菜单，然后在下拉菜单的选项中添加 **.dropdown-item** 类。我们也可以在 **<a>** 标签中使用

.dropdown-divider 类用于在下拉菜单中创建一个水平的分割线： `<div class="dropdown-divider"></div>`

Carousel 轮播效果

通过 `.carousel` 命名样式引入轮播组件，同时为此控件设置唯一的ID-尤其是当你在同一页面使用多个轮播效果时这是必须的。

```
<div id="carouselExampleControls" class="carousel slide" data-ride="carousel">
  <div class="carousel-inner">
    <div class="carousel-item active">
      
    </div>
    <div class="carousel-item">
      
    </div>
    <div class="carousel-item">
      
    </div>
  </div>
  <a class="carousel-control-prev" href="#carouselExampleControls" role="button" data-slide="prev">
    <span class="carousel-control-prev-icon" aria-hidden="true"></span>
    <span class="sr-only">Previous</span>
  </a>
  <a class="carousel-control-next" href="#carouselExampleControls" role="button" data-slide="next">
    <span class="carousel-control-next-icon" aria-hidden="true"></span>
    <span class="sr-only">Next</span>
  </a>
</div>
```

`.d-block`、`.w-100`两个样式，以修正浏览器预设的图像对齐带来的影响。

Carousel 轮播效果

在每个 `<div class="carousel-item">` 内添加 `<div class="carousel-caption">` 来设置轮播图片的描述文本

```
<div class="carousel-item">
  
  <div class="carousel-caption">
    <h3>第一张图片描述标题</h3>
    <p>描述文字!</p>
  </div>
</div>
```

Carousel 轮播效果

轮播的其他效果实现类

Class	Description
<code>.carousel</code>	创建一个轮播
<code>.carousel-indicators</code>	为轮播添加一个指示符，就是轮播图底下的一个个小点，轮播的过程可以显示目前是第几张图。
<code>.carousel-inner</code>	添加要切换的图片
<code>.carousel-item</code>	指定每个图片的内容
<code>.carousel-control-prev</code>	添加左侧的按钮，点击会返回上一张。
<code>.carousel-control-next</code>	添加右侧的按钮，点击会返回下一张。
<code>.carousel-control-prev-icon</code>	与 <code>.carousel-control-prev</code> 一起使用，设置左侧的按钮
<code>.carousel-control-next-icon</code>	与 <code>.carousel-control-next</code> 一起使用，设置右侧的按钮

Buttons & Button group 按钮系列

Bootstrap 4 提供了不同样式的按钮

```
<button type="button" class="btn btn-primary">Primary</button>
<button type="button" class="btn btn-secondary">Secondary</button>
<button type="button" class="btn btn-success">Success</button>
<button type="button" class="btn btn-danger">Danger</button>
<button type="button" class="btn btn-warning">Warning</button>
<button type="button" class="btn btn-info">Info</button>
<button type="button" class="btn btn-light">Light</button>
<button type="button" class="btn btn-dark">Dark</button>
<button type="button" class="btn btn-link">Link</button>
```

- 轮廓按钮：用默认修饰符类替换 `.btn-outline-*` 任何按钮上的所有背景颜色和图像
- 尺寸大小：配合 `.btn-lg` 、 `.btn-sm` 两个邻近元素，可分别实现大、小规格按钮
- 启用状态： `.btn` 样式定义的按钮默认启用状态，如果一定要使按钮固定为启用状态、不需要点击反馈，可以增加 `.active` 样式，并包括 `aria-pressed="true"` 属性
- 禁用状态：直接将 `disabled` 布尔属性添加到任何 `<button>` 元素

Forms & Input 表单&输入框

Bootstrap 通过一些简单的 HTML 标签和扩展的类即可创建出不同样式的表单
表单元素 `<input>`, `<textarea>`, 和 `<select>` elements 在使用 `.form-control` 类的情况下,
宽度都是设置为 100%

两种表单布局：堆叠表单-垂直方向；内联表单-水平方向

`<!--这是堆叠表单 -->`

```
<form>
  <div class="form-group">
    <label for="email">Email address:</label>
    <input type="email" class="form-control" id="email">
  </div>
  <div class="form-group">
    <label for="pwd">Password:</label>
    <input type="password" class="form-control" id="pwd">
  </div>
  <div class="form-check">
    <label class="form-check-label">
      <input class="form-check-input" type="checkbox"> Remember me
    </label>
  </div>
  <button type="submit" class="btn btn-primary">Submit</button>
</form>
```


Forms & Input 表单&输入框

内联表单需要在 <form> 元素上添加 .form-inline类。

注意：在屏幕宽度小于 576px 时为垂直堆叠，如果屏幕宽度大于等于576px时表单元素才会显示在同一个水平线上。

<!这是内联表单 -->

```
<form class="form-inline">
  <label for="email">Email address:</label>
  <input type="email" class="form-control" id="email">
  <label for="pwd">Password:</label>
  <input type="password" class="form-control" id="pwd">
  <div class="form-check">
    <label class="form-check-label">
      <input class="form-check-input" type="checkbox"> Remember me
    </label>
  </div>
  <button type="submit" class="btn btn-primary">Submit</button>
</form>
```

Forms & Input 表单&输入框

Bootstrap 4支持以下几种表单控件：

input 输入

textarea 文本框

checkbox 复选框

radio 单选框

select 下拉菜单

对于input文件选择控件，Bootstrap v4采用`.form-control-file`取代了`.form-control`。

```
<form>
  <div class="form-group">
    <label for="exampleFormControlFile1">Example file input</label>
    <input type="file" class="form-control-file" id="exampleFormControlFile1">
  </div>
</form>
```

Input实例

```
<div class="form-group">
  <label for="usr">用户名:</label>
  <input type="text" class="form-control" id="usr">
</div>
<div class="form-group">
  <label for="pwd">密码:</label>
  <input type="password" class="form-control" id="pwd">
</div>
```

Textarea实例

```
<div class="form-group">
  <label for="comment">评论:</label>
  <textarea class="form-control" rows="5" id="comment"></textarea>
</div>
```

Card 卡片

`.card` 卡片组件是Bootstrap 4新增的一组重要样式，它是一个灵活的、可扩展的内容容器，包含了可选的卡片头和卡片脚、一个大范围的内容、上下文背景色以及强大的显示选项。

通过 `.card` 与 `.card-body` 类就可以创建一个最简单的卡片：

```
<div class="container">
  <h2>简单的卡片</h2>
  <div class="card">
    <div class="card-body">简单的卡片</div>
  </div>
</div>
```

来加点图片和按钮吧！

```
<div class="card" style="width: 18rem;">
  
  <div class="card-body">
    <h5 class="card-title">Card title</h5>
    <p class="card-text">Some quick example text to build on the card title and make up the bulk of the card's content.</p>
    <a href="#" class="btn btn-primary">Go somewhere</a>
  </div>
</div>
```

Card 卡片

卡片与栅格系统结合使用

使用栅格系统，根据需求按行与列来装载卡片。

```
<div class="row">
  <div class="col-sm-6">
    <div class="card">
      <div class="card-body">
        <h5 class="card-title">Special title treatment</h5>
        <p class="card-text">With supporting text below as a natural lead-in to
additional content.</p>
        <a href="#" class="btn btn-primary">Go somewhere</a>
      </div>
    </div>
  </div>
  <div class="col-sm-6">
    <div class="card">
      <div class="card-body">
        <h5 class="card-title">Special title treatment</h5>
        <p class="card-text">With supporting text below as a natural lead-in to
additional content.</p>
        <a href="#" class="btn btn-primary">Go somewhere</a>
      </div>
    </div>
  </div>
</div>
```

Media-object 媒体对象

Bootstrap 提供了很好的方式来处理多媒体对象（图片或视频）和内容的布局。应用场景有博客评论、微博等：

以下是一个媒体对象的示例，只需要两个Class选择器，分别引用`.media`和`.media-body`来装载的内容，就可以实现页面设计目标、形成布局、间距并控制可选的填充和边距。

```
<div class="media">
  
  <div class="media-body">
    <h5 class="mt-0">Media heading</h5>
    Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque
    ante sollicitudin. Cras purus odio, vestibulum in vulputate at, tempus
    viverra turpis. Fusce condimentum nunc ac nisi vulputate fringilla. Donec
    lacinia congue felis in faucibus.
  </div>
</div>
```

- 嵌套、对齐、排序、列表

Alert 警告提示

Bootstrap 4

Modal 弹出模态框

Bootstrap 4

Part 4

Utilities 公共样式

- Border 边框
- Embed 嵌入
- Position 固顶/底以及定位
- Shadows 阴影
- Sizi 规格
- Text 文本处理

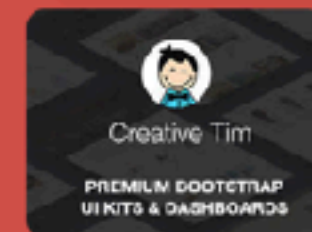


To be continued...

Bootstrap templates & themes

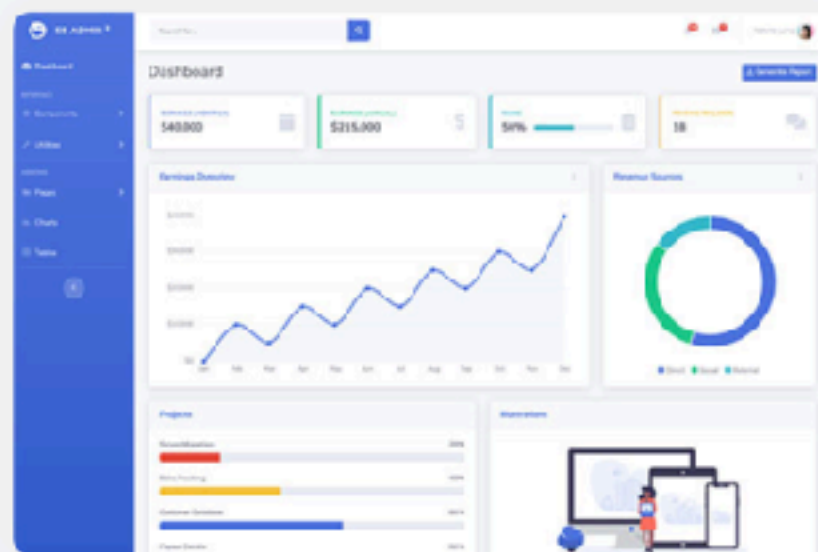
Bootstrap Templates & Themes

Free Bootstrap themes that are ready to customize and publish. All of our themes are built with Bootstrap 4, MIT licensed, and updated regularly!



Premium Bootstrap themes, templates, and UI kits by Creative Tim.

ads via Carbon



SB Admin 2

A free Bootstrap admin theme

Free



SB UI Kit Pro

A premium Bootstrap 4 UI Kit

Pro



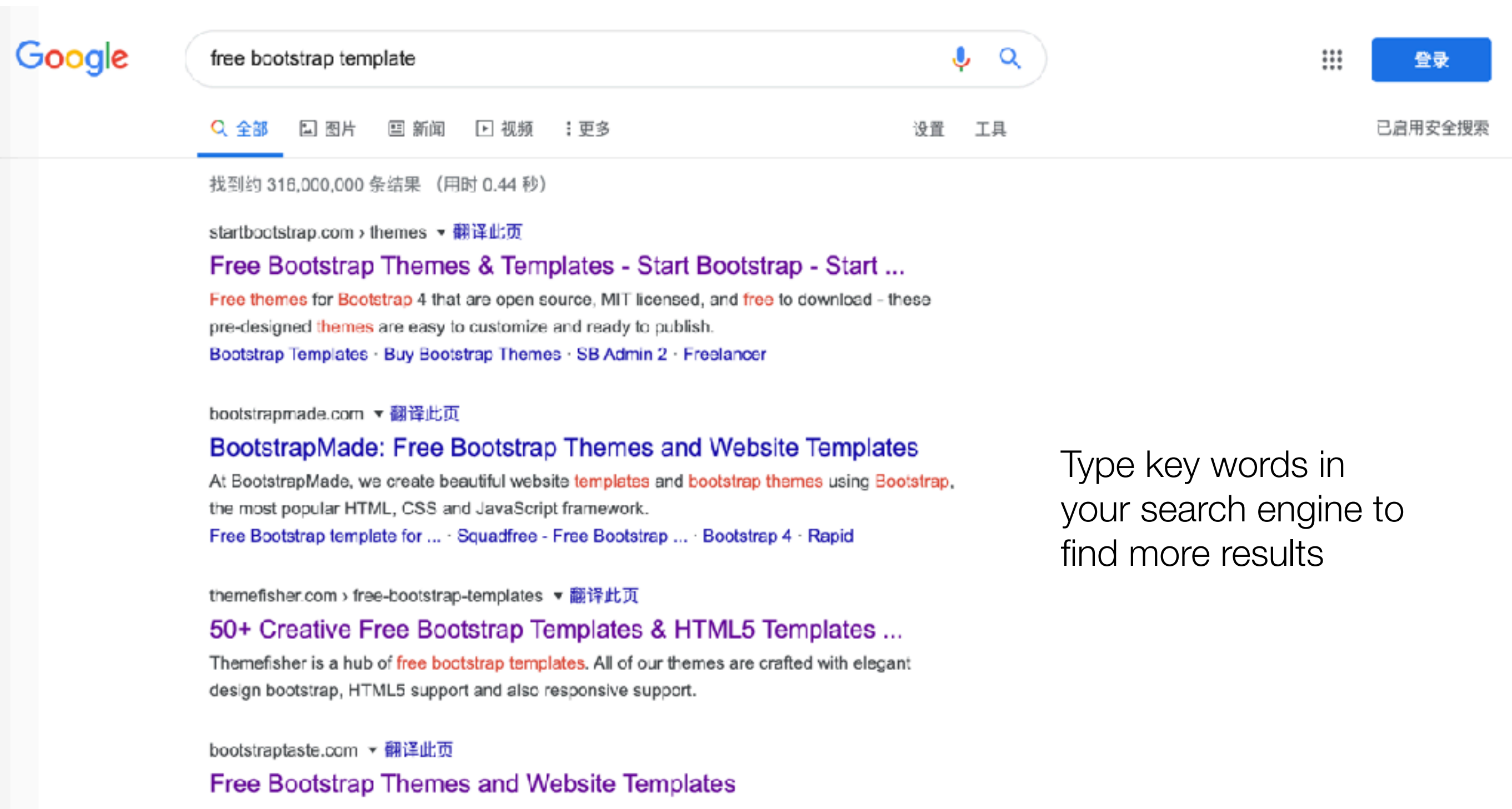
SB Admin Pro

A premium Bootstrap admin theme

Pro

<https://startbootstrap.com/themes/>

Bootstrap templates & themes



The screenshot shows a Google search interface with the query "free bootstrap template". The search results are displayed in a list format. The first result is from startbootstrap.com, titled "Free Bootstrap Themes & Templates - Start Bootstrap - Start ...". The second result is from bootstrapmade.com, titled "BootstrapMade: Free Bootstrap Themes and Website Templates". The third result is from themefisher.com, titled "50+ Creative Free Bootstrap Templates & HTML5 Templates ...". The fourth result is from bootstraptaste.com, titled "Free Bootstrap Themes and Website Templates".

Google

free bootstrap template

登录

全部 图片 新闻 视频 更多 设置 工具 已启用安全搜索

找到约 316,000,000 条结果 (用时 0.44 秒)

startbootstrap.com > themes ▾ 翻译此页

Free Bootstrap Themes & Templates - Start Bootstrap - Start ...

Free themes for Bootstrap 4 that are open source, MIT licensed, and free to download - these pre-designed themes are easy to customize and ready to publish.

Bootstrap Templates · Buy Bootstrap Themes · SB Admin 2 · Freelancer

bootstrapmade.com ▾ 翻译此页

BootstrapMade: Free Bootstrap Themes and Website Templates

At BootstrapMade, we create beautiful website templates and bootstrap themes using Bootstrap, the most popular HTML, CSS and JavaScript framework.

Free Bootstrap template for ... · Squadfree - Free Bootstrap ... · Bootstrap 4 · Rapid

themefisher.com > free-bootstrap-templates ▾ 翻译此页

50+ Creative Free Bootstrap Templates & HTML5 Templates ...

Themefisher is a hub of free bootstrap templates. All of our themes are crafted with elegant design bootstrap, HTML5 support and also responsive support.

bootstraptaste.com ▾ 翻译此页

Free Bootstrap Themes and Website Templates

Type key words in
your search engine to
find more results

Thanks for your time!