

# 栅格系统

## 布局容器

Bootstrap 需要为页面内容和栅格系统包裹一个 `.container` 容器。

### `.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>
```

注意，这两种 容器类不能互相嵌套。

## 栅格参数

	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			
.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				

## 行与列

- “行（row）”必须包含在 `.container`（固定宽度）或 `.container-fluid`（100% 宽度）中

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

- 只有“列（column）”可以作为行（row）”的直接子元素。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。

```
<div class="row">
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
</div>
<div class="row">
  <div class="col-md-8">.col-md-8</div>
  <div class="col-md-4">.col-md-4</div>
</div>
<div class="row">
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4">.col-md-4</div>
</div>
<div class="row">
  <div class="col-md-6">.col-md-6</div>
  <div class="col-md-6">.col-md-6</div>
</div>
```

.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1
.col-md-8								.col-md-4			
.col-md-4				.col-md-4				.col-md-4			
.col-md-6						.col-md-6					

- 如果一行（row）”中包含的“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。

```
<div class="row">
  <div class="col-xs-9">.col-xs-9</div>
  <div class="col-xs-4">.col-xs-4<br>Since 9 + 4 = 13 > 12, this 4-column-wide div gets wrapped
```

onto a new line as one contiguous unit.</div>

```
<div class="col-xs-6">.col-xs-6<br>Subsequent columns continue along the new line.</div>
</div>
```

.col-xs-9	
.col-xs-4 Since $9 + 4 = 13 > 12$ , this 4-column-wide div gets wrapped onto a new line as one contiguous unit.	.col-xs-6 Subsequent columns continue along the new line.

## 移动设备和桌面屏幕

针对超小屏幕和中等屏幕设备，一起使用 `.col-xs-*` 和 `.col-md-*`。

<!-- Stack the columns on mobile by making one full-width and the other half-width -->

```
<div class="row">
  <div class="col-xs-12 col-md-8">.col-xs-12 .col-md-8</div>
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>
</div>
```

<!-- Columns start at 50% wide on mobile and bump up to 33.3% wide on desktop -->

```
<div class="row">
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>
</div>
```

<!-- Columns are always 50% wide, on mobile and desktop -->

```
<div class="row">
  <div class="col-xs-6">.col-xs-6</div>
  <div class="col-xs-6">.col-xs-6</div>
</div>
```

.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-6		.col-xs-6	

## 响应式列重置

某些列可能会出现比别的列高的情况，建议使用 `.clearfix`

## 列偏移

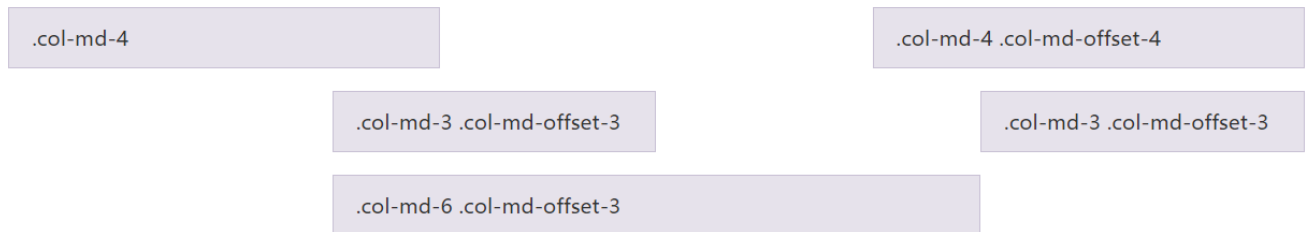
`.col-md-offset-*` 类可以将列向右侧偏移。\* 表示偏移的列数。

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

```

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

```



## 嵌套列

添加一个新的 `.row` 元素和一系列 `.col-sm-*` 元素到已经存在的 `.col-sm-*` 元素内。被嵌套的行所包含的列的个数不能超过12。

```

<div class="row">
  <div class="col-sm-9">
    Level 1: .col-sm-9
    <div class="row">
      <div class="col-xs-8 col-sm-6">
        Level 2: .col-xs-8 .col-sm-6
      </div>
      <div class="col-xs-4 col-sm-6">
        Level 2: .col-xs-4 .col-sm-6
      </div>
    </div>
  </div>
</div>

```



## 列排序

通过使用 `.col-md-push-*` 和 `.col-md-pull-*` 类就可以改变列的顺序。

```

<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>

```



## 列间隔

- 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。