



# Bootstrap 5

---

## First Look



# Bootstrap e seu sistema de Grids

**O Bootstrap pode parecer um pouco complicado no início, mas rapidamente você pega prática e entende o sistema de grids facilmente.**

**Basicamente, o Bootstrap funciona no sistema de containers, com linhas e colunas.**

# Bootstrap e seu sistema de Grids

**Cada linha pode possuir até 12 colunas. E, essas linhas, devem estar dentro de containers e as colunas dentro de linhas.**

1 col-1	2	3	4	5	6	7	8	9	10	11	12
------------	---	---	---	---	---	---	---	---	----	----	----

Row

col-6	col-6
-------	-------

Row

col-12	<div class="col-sm-12 col-lg-4"> </div>									
col-4	col-4	col-4	col-4	col-4	col-4	col-4	col-4	col-4	col-4	col-4

Row

**Colunas:**

Extra small	Small	Medium	Large	Extra large
<576px	≥576px	≥768px	≥992px	≥1200px
.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-

Container

# entendendo o tamanhos das containers

	<b>Extra small</b> <576px	<b>Small</b> ≥576px	<b>Medium</b> ≥768px	<b>Large</b> ≥992px	<b>X-Large</b> ≥1200px	<b>XX-Large</b> ≥1400px
<code>.container</code>	100%	540px	720px	960px	1140px	1320px
<code>.container-sm</code>	100%	540px	720px	960px	1140px	1320px
<code>.container-md</code>	100%	100%	720px	960px	1140px	1320px
<code>.container-lg</code>	100%	100%	100%	960px	1140px	1320px
<code>.container-xl</code>	100%	100%	100%	100%	1140px	1320px
<code>.container-xxl</code>	100%	100%	100%	100%	100%	1320px
<code>.container-fluid</code>	100%	100%	100%	100%	100%	100%

# entendendo o tamanhos das containers

```
<div class="container">
  <!-- Content here -->
</div>
```

[Copy](#)

## Responsive containers

Responsive containers allow you to specify a class that is 100% wide until the specified breakpoint is reached, after which we apply `max-widths` for each of the higher breakpoints. For example, `.container-sm` is 100% wide to start until the `sm` breakpoint is reached, where it will scale up with `md`, `lg`, `xl`, and `xxl`.

```
<div class="container-sm">100% wide until small breakpoint</div>
<div class="container-md">100% wide until medium breakpoint</div>
<div class="container-lg">100% wide until large breakpoint</div>
<div class="container-xl">100% wide until extra large breakpoint</div>
<div class="container-xxl">100% wide until extra extra large breakpoint</div>
```

[Copy](#)

# Testando ao Containers

```
51 <div class="container">
52   <div class="row">
53     <div class="col border bg-info">
54       Coluna
55     </div>
56     <div class="col border bg-info">
57       Coluna
58     </div>
59   </div>
60 </div>
```





COLUNAS

**comportamentos dos tamanhos das  
colunas**



# TRABALHANDO COM MARGENS

> Getting started

> Customize

> Layout

> Content

> Forms

> Components

> Helpers

▼ Utilities

API

Background

Borders

Colors

Display

Flex

Float

Interactions

Overflow

Position

Shadows

## Spacing

[View on GitHub](#)

Bootstrap includes a wide range of shorthand responsive margin, padding, and gap utility classes to modify an element's appearance.

### Margin and padding

Assign responsive-friendly `margin` or `padding` values to an element or a subset of its sides with shorthand classes. Includes support for individual properties, all properties, and vertical and horizontal properties. Classes are built from a default Sass map ranging from `.25rem` to `3rem`.

Using the CSS Grid layout module? Consider using [the gap utility](#).

### Notation

Spacing utilities that apply to all breakpoints, from `xs` to `xxl`, have no breakpoint abbreviation in them. This is because those classes are applied from `min-width: 0` and up, and thus are not bound by a media query. The remaining breakpoints, however, do include a breakpoint abbreviation.

The classes are named using the format `{property}{sides}-{size}` for `xs` and `{property}{sides}-{breakpoint}-{size}` for `sm`, `md`, `lg`, `xl`, and `xxl`.

Where *property* is one of:

# TRABALHANDO COM MARGENS

Where *property* is one of:

- `m` - for classes that set `margin`
- `p` - for classes that set `padding`

**m** - setar a margem

**p** - setar o Padding

# TRABALHANDO COM MARGENS

Where *sides* is one of:

- **t** - for classes that set `margin-top` or `padding-top`
- **b** - for classes that set `margin-bottom` or `padding-bottom`
- **s** - (start) for classes that set `margin-left` or `padding-left` in LTR, `margin-right` or `padding-right` in RTL
- **e** - (end) for classes that set `margin-right` or `padding-right` in LTR, `margin-left` or `padding-left` in RTL
- **x** - for classes that set both `*-left` and `*-right`
- **y** - for classes that set both `*-top` and `*-bottom`
- **blank** - for classes that set a `margin` or `padding` on all 4 sides of the element

# TRABALHANDO COM MARGENS

Where *size* is one of:

- `0` - for classes that eliminate the `margin` or `padding` by setting it to `0`
- `1` - (by default) for classes that set the `margin` or `padding` to `$spacer * .25`
- `2` - (by default) for classes that set the `margin` or `padding` to `$spacer * .5`
- `3` - (by default) for classes that set the `margin` or `padding` to `$spacer`
- `4` - (by default) for classes that set the `margin` or `padding` to `$spacer * 1.5`
- `5` - (by default) for classes that set the `margin` or `padding` to `$spacer * 3`
- `auto` - for classes that set the `margin` to `auto`

# TRABALHANDO COM MARGENS

## Horizontal centering

Additionally, Bootstrap also includes an `.mx-auto` class for horizontally centering fixed-width block level content—that is, content that has `display: block` and a `width` set—by setting the horizontal margins to `auto`.

Centered element

```
<div class="mx-auto" style="width: 200px;">  
  Centered element  
</div>
```

Copy