

[View on GitHub](#)

Columns

Learn how to modify columns with a handful of options for alignment, ordering, and offsetting thanks to our flexbox grid system. Plus, see how to use column classes to manage widths of non-grid elements.



See what your team could do with The DevSecOps Platform. Take GitLab for a spin.

ads via Carbon

On this page

How they work

Alignment

Vertical alignment

Horizontal alignment

Column wrapping

Column breaks

Reordering

Order classes

Offsetting columns

Offset classes

Margin utilities

Standalone column classes

Heads up! Be sure to [read the Grid page](#) first before diving into how to modify and customize your grid columns.

How they work

- **Columns build on the grid's flexbox architecture.** Flexbox means we have options for changing individual columns and [modifying groups of columns at the row level](#). You choose how columns grow, shrink, or otherwise change.
- **When building grid layouts, all content goes in columns.** The hierarchy of Bootstrap's grid goes from [container](#) to row to column to your content. On rare occasions, you may combine content and column, but be aware there can be unintended consequences.
- **Bootstrap includes predefined classes for creating fast, responsive layouts.** With [six breakpoints](#) and a dozen columns at each grid tier, we have dozens of classes already built for you to create your desired layouts. This can be disabled via Sass if you wish.

Alignment

Use flexbox alignment utilities to vertically and horizontally align columns.

Vertical alignment

One of three columns

One of three columns

One of three columns

One of three columns

One of three columns

One of three columns

One of three columns

One of three columns

One of three columns

```
<div class="container">
  <div class="row align-items-start">
    <div class="col">
      One of three columns
    </div>
    <div class="col">
      One of three columns
    </div>
    <div class="col">
      One of three columns
    </div>
  </div>
  <div class="row align-items-center">
    <div class="col">
      One of three columns
    </div>
    <div class="col">
      One of three columns
    </div>
    <div class="col">
      One of three columns
    </div>
  </div>
  <div class="row align-items-end">
    <div class="col">
      One of three columns
    </div>
    <div class="col">
      One of three columns
    </div>
    <div class="col">
      One of three columns
    </div>
  </div>
</div>
```

One of three columns

One of three columns

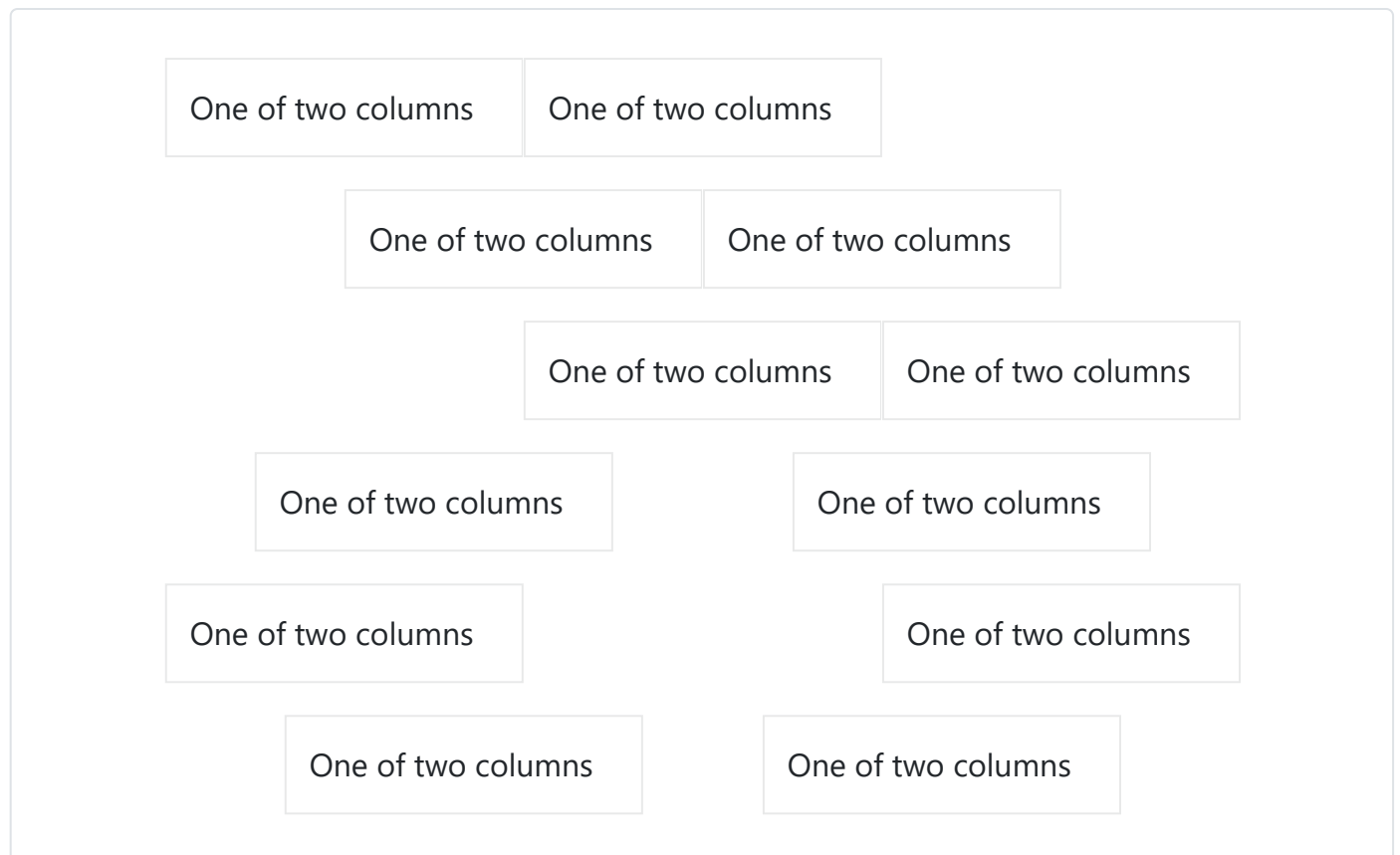
One of three columns

```

<div class="container">
  <div class="row">
    <div class="col align-self-start">
      One of three columns
    </div>
    <div class="col align-self-center">
      One of three columns
    </div>
    <div class="col align-self-end">
      One of three columns
    </div>
  </div>
</div>

```

Horizontal alignment



```

<div class="container">
  <div class="row justify-content-start">
    <div class="col-4">
      One of two columns
    </div>
    <div class="col-4">
      One of two columns
    </div>
  </div>

```

```

</div>
<div class="row justify-content-center">
  <div class="col-4">
    One of two columns
  </div>
  <div class="col-4">
    One of two columns
  </div>
</div>
<div class="row justify-content-end">
  <div class="col-4">
    One of two columns
  </div>
  <div class="col-4">
    One of two columns
  </div>
</div>
<div class="row justify-content-around">
  <div class="col-4">
    One of two columns
  </div>
  <div class="col-4">
    One of two columns
  </div>
</div>
<div class="row justify-content-between">
  <div class="col-4">
    One of two columns
  </div>
  <div class="col-4">
    One of two columns
  </div>
</div>
<div class="row justify-content-evenly">
  <div class="col-4">
    One of two columns
  </div>
  <div class="col-4">
    One of two columns
  </div>
</div>
</div>

```

Column wrapping

If more than 12 columns are placed within a single row, each group of extra columns will, as one unit, wrap onto a new line.

.col-9

.col-4

Since $9 + 4 = 13 > 12$, this 4-column-wide div gets wrapped onto a new line as one contiguous unit.

.col-6

Subsequent columns continue along the new line.

```
<div class="container">
  <div class="row">
    <div class="col-9">.col-9</div>
    <div class="col-4">.col-4<br>Since 9 + 4 = 13 > 12, this 4-column-wide div ge
    <div class="col-6">.col-6<br>Subsequent columns continue along the new line.</di
  </div>
</div>
```

Column breaks

Breaking columns to a new line in flexbox requires a small hack: add an element with `width: 100%` wherever you want to wrap your columns to a new line. Normally this is accomplished with multiple `.rows`, but not every implementation method can account for this.

.col-6 .col-sm-3

.col-6 .col-sm-3

.col-6 .col-sm-3

.col-6 .col-sm-3

```
<div class="container">
  <div class="row">
    <div class="col-6 col-sm-3">.col-6 .col-sm-3</div>
    <div class="col-6 col-sm-3">.col-6 .col-sm-3</div>

    <!-- Force next columns to break to new line -->
    <div class="w-100"></div>
```

```

<div class="col-6 col-sm-3">.col-6 .col-sm-3</div>
<div class="col-6 col-sm-3">.col-6 .col-sm-3</div>
</div>
</div>

```

You may also apply this break at specific breakpoints with our [responsive display utilities](#).

| | | |
|------------------|------------------|------------------|
| .col-6 .col-sm-4 | .col-6 .col-sm-4 | .col-6 .col-sm-4 |
| .col-6 .col-sm-4 | | |

```

<div class="container">
  <div class="row">
    <div class="col-6 col-sm-4">.col-6 .col-sm-4</div>
    <div class="col-6 col-sm-4">.col-6 .col-sm-4</div>

    <!-- Force next columns to break to new line at md breakpoint and up -->
    <div class="w-100 d-none d-md-block"></div>

    <div class="col-6 col-sm-4">.col-6 .col-sm-4</div>
    <div class="col-6 col-sm-4">.col-6 .col-sm-4</div>
  </div>
</div>

```

Reordering

Order classes

Use **.order-** classes for controlling the **visual order** of your content. These classes are responsive, so you can set the **order** by breakpoint (e.g., **.order-1.order-md-2**). Includes support for **1** through **5** across all six grid tiers.

| | | |
|--------------------------------|----------------------------------|------------------------------------|
| First in DOM, no order applied | Third in DOM, with an order of 1 | Second in DOM, with a larger order |
|--------------------------------|----------------------------------|------------------------------------|

```

<div class="container">
  <div class="row">
    <div class="col">
      First in DOM, no order applied
    </div>
    <div class="col order-5">
      Second in DOM, with a larger order
    </div>
    <div class="col order-1">
      Third in DOM, with an order of 1
    </div>
  </div>
</div>

```

There are also responsive `.order-first` and `.order-last` classes that change the `order` of an element by applying `order: -1` and `order: 6`, respectively. These classes can also be intermixed with the numbered `.order-*` classes as needed.

| | | |
|--------------------------------|-----------------------------|-------------------------------|
| Third in DOM, ordered first | Second in DOM, unordered | First in DOM, ordered last |
|--------------------------------|-----------------------------|-------------------------------|

```

<div class="container">
  <div class="row">
    <div class="col order-last">
      First in DOM, ordered last
    </div>
    <div class="col">
      Second in DOM, unordered
    </div>
    <div class="col order-first">
      Third in DOM, ordered first
    </div>
  </div>
</div>

```

Offsetting columns

You can offset grid columns in two ways: our responsive `.offset-` grid classes and our [margin utilities](#). Grid classes are sized to match columns while margins are more useful for quick layouts where the width of the offset is variable.

Offset classes

Move columns to the right using `.offset-md-*` classes. These classes increase the left margin of a column by `*` columns. For example, `.offset-md-4` moves `.col-md-4` over four columns.

`.col-md-4`

`.col-md-4 .offset-md-4`

`.col-md-3 .offset-md-3`

`.col-md-3 .offset-md-3`

`.col-md-6 .offset-md-3`

```
<div class="container">
  <div class="row">
    <div class="col-md-4">.col-md-4</div>
    <div class="col-md-4 offset-md-4">.col-md-4 .offset-md-4</div>
  </div>
  <div class="row">
    <div class="col-md-3 offset-md-3">.col-md-3 .offset-md-3</div>
    <div class="col-md-3 offset-md-3">.col-md-3 .offset-md-3</div>
  </div>
  <div class="row">
    <div class="col-md-6 offset-md-3">.col-md-6 .offset-md-3</div>
  </div>
</div>
```

In addition to column clearing at responsive breakpoints, you may need to reset offsets. See this in action in [the grid example](#).

`.col-sm-5 .col-md-6`

`.col-sm-5 .offset-sm-2 .col-md-6 .offset-md-0`

`.col-sm-6 .col-md-5 .col-lg-6`

`.col-sm-6 .col-md-5 .offset-md-2 .col-lg-6 .offset-lg-0`

```

<div class="container">
  <div class="row">
    <div class="col-sm-5 col-md-6">.col-sm-5 .col-md-6</div>
    <div class="col-sm-5 offset-sm-2 col-md-6 offset-md-0">.col-sm-5 .offset-sm-2 .c
  </div>
  <div class="row">
    <div class="col-sm-6 col-md-5 col-lg-6">.col-sm-6 .col-md-5 .col-lg-6</div>
    <div class="col-sm-6 col-md-5 offset-md-2 col-lg-6 offset-lg-0">.col-sm-6 .col-m
  </div>
</div>

```

Margin utilities

With the move to flexbox in v4, you can use margin utilities like `.me-auto` to force sibling columns away from one another.

`.col-md-4`

`.col-md-4 .ms-auto`

`.col-md-3 .ms-md-auto`

`.col-md-3 .ms-md-auto`

`.col-auto .me-auto`

`.col-auto`

```

<div class="container">
  <div class="row">
    <div class="col-md-4">.col-md-4</div>
    <div class="col-md-4 ms-auto">.col-md-4 .ms-auto</div>
  </div>
  <div class="row">
    <div class="col-md-3 ms-md-auto">.col-md-3 .ms-md-auto</div>
    <div class="col-md-3 ms-md-auto">.col-md-3 .ms-md-auto</div>
  </div>
  <div class="row">
    <div class="col-auto me-auto">.col-auto .me-auto</div>
    <div class="col-auto">.col-auto</div>
  </div>
</div>

```

```
</div>  
</div>
```

Standalone column classes

The `.col-*` classes can also be used outside a `.row` to give an element a specific width. Whenever column classes are used as non direct children of a row, the paddings are omitted.

`.col-3`: width of 25%

`.col-sm-9`: width of 75% above sm breakpoint

```
<div class="col-3 bg-light p-3 border">  
  .col-3: width of 25%  
</div>  
<div class="col-sm-9 bg-light p-3 border">  
  .col-sm-9: width of 75% above sm breakpoint  
</div>
```

The classes can be used together with utilities to create responsive floated images. Make sure to wrap the content in a [.clearfix](#) wrapper to clear the float if the text is shorter.

Responsive floated image

A paragraph of placeholder text. We're using it here to show the use of the `clearfix` class. We're adding quite a few meaningless phrases here to demonstrate how the columns interact here with the floated image.

As you can see the paragraphs gracefully wrap around the floated image. Now imagine how this would look with some actual content in here, rather than just this boring placeholder text that goes on and on, but actually conveys no tangible information at. It simply takes up space and should not really be read.

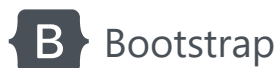

And yet, here you are, still persevering in reading this placeholder text, hoping for some more insights, or some hidden easter egg of content. A joke, perhaps. Unfortunately, there's none of that here.

```
<div class="clearfix">
  

  <p>
    A paragraph of placeholder text. We're using it here to show the use of the clea
  </p>

  <p>
    As you can see the paragraphs gracefully wrap around the floated image. Now imag
  </p>

  <p>
    And yet, here you are, still persevering in reading this placeholder text, hopin
  </p>
</div>
```



Designed and built with all the love in the world by the Bootstrap team with the help of our contributors.

Code licensed MIT, docs CC BY 3.0.

Currently v5.0.2.

Analytics by Fathom.

Links

[Home](#)

[Docs](#)

Guides

[Getting started](#)

[Starter template](#)

[Examples](#)

[Themes](#)

[Blog](#)

[Swag Store](#)

Projects

[Bootstrap 5](#)

[Bootstrap 4](#)

[Icons](#)

[RFS](#)

[npm starter](#)

[Webpack](#)

[Parcel](#)

Community

[Issues](#)

[Discussions](#)

[Corporate sponsors](#)

[Open Collective](#)

[Stack Overflow](#)