

View on GitHub

Containers

Containers are a fundamental building block of Bootstrap that contain, pad, and align your content within a given device or viewport.



Get all you need in one database, including JSON, SQL, storage, search, and analytics.
ads via Carbon

On this page

◇

How they work

Containers are the most basic layout element in Bootstrap and are **required when using our default grid system**. Containers are used to contain, pad, and (sometimes) center the content within them. While containers *can* be nested, most layouts do not require a nested container.

Bootstrap comes with three different containers:

- `.container`, which sets a `max-width` at each responsive breakpoint
- `.container-{breakpoint}`, which is `width: 100%` until the specified breakpoint
- `.container-fluid`, which is `width: 100%` at all breakpoints

The table below illustrates how each container’s `max-width` compares to the original `.container` and `.container-fluid` across each breakpoint.

See them in action and compare them in our [Grid example](#).

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

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

Default container

Our default `.container` class is a responsive, fixed-width container, meaning its `max-width` changes at each breakpoint.

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

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>
```

Fluid containers

Use `.container-fluid` for a full width container, spanning the entire width of the viewport.

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

CSS

Sass variables

As shown above, Bootstrap generates a series of predefined container classes to help you build the layouts you desire. You may customize these predefined container classes by modifying the Sass map (found in `_variables.scss`) that powers them:

scss/_variables.scss



```
$container-max-widths: (
  sm: 540px,
  md: 720px,
  lg: 960px,
  xl: 1140px,
  xxl: 1320px
);
```

For more information and examples on how to modify our Sass maps and variables, please refer to [the Sass section of the Grid documentation](#).

Sass mixins

In addition to customizing the Sass, you can also create your own containers with our Sass mixin.

```
// Source mixin
@mixin make-container($padding-x: $container-padding-x) {
  width: 100%;
  padding-right: $padding-x;
  padding-left: $padding-x;
  margin-right: auto;
  margin-left: auto;
}
```

// Usage

```
.custom-container {  
  @include make-container();  
}
```

Bootstrap

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 v5.3.0.

Links

- Home
- Docs
- Examples
- Icons
- Themes
- Blog
- Swag Store

Projects

- Bootstrap 5
- Bootstrap 4
- Icons
- RFS
- Examples repo

Guides

- Getting started
- Starter template
- Webpack
- Parcel
- Vite

Community

- Issues
- Discussions
- Corporate sponsors
- Open Collective
- Stack Overflow