Spacing

Contents

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:

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

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

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

(You can add more sizes by adding entries to the `$spacers` Sass map variable.)

Examples

Here are some representative examples of these classes:

```
.mt-0 {
  margin-top: 0 !important;
}

.ms-1 {
  margin-left: ($spacer * .25) !important;
}

.px-2 {
  padding-left: ($spacer * .5) !important;
  padding-right: ($spacer * .5) !important;
}

.p-3 {
  padding: $spacer !important;
}
```

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>
```

Negative margin

In CSS, `margin` properties can utilize negative values (`padding` cannot). These negative margins are **disabled by default**, but can be enabled in Sass by setting `$enable-negative-margins: true`.

The syntax is nearly the same as the default, positive margin utilities, but with the addition of `n` before the requested size. Here's an example class that's the opposite of `.mt-1`:

```
.mt-n1 {
  margin-top: -0.25rem !important;
}
```

Gap

When using `display: grid`, you can make use of `gap` utilities on the parent grid container. This can save on having to add margin utilities to individual grid items (children of a `display: grid` container). Gap utilities are responsive by default, and are generated via our utilities API, based on the `$spacers` Sass map.

{{< example html >}}

Grid item 1

Grid item 2

Grid item 3

{{< /example >}}

Support includes responsive options for all of Bootstrap's grid breakpoints, as well as six sizes from the `$spacers` map (`0`–`5`). There is no `.gap-auto` utility class as it's effectively the same as `.gap-0`.

Sass

Maps

Spacing utilities are declared via Sass map and then generated with our utilities API.

{{< scss-docs name="spacer-variables-maps" file="scss/_variables.scss" >}}

Utilities API

Spacing utilities are declared in our utilities API in `scss/_utilities.scss`. [Learn how to use the utilities API.]({{< docsref "/utilities/api#using-the-api" >}})

{{< scss-docs name="utils-spacing" file="scss/_utilities.scss" >}}