

Migrating to v5

Contents

Dependencies	2
Browser support	2
Documentation changes	2
Sass	2
Color system	4
Grid updates	4
Content, Reboot, etc	5
RTL	6
Forms	6
Components	6
Accordion	7
Alerts	7
Badges	7
Breadcrumbs	7
Buttons	7
Card	8
Carousel	8
Close button	8
Collapse	8
Dropdowns	8
Jumbotron	9
List group	9
Navs and tabs	9
Navbars	9
Offcanvas	9
Pagination	9
Popovers	9
Spinners	10
Toasts	10
Tooltips	10
Utilities	10
Helpers	11
JavaScript	12

Dependencies

- Dropped jQuery.
- Upgraded from Popper v1.x to Popper v2.x.
- Replaced Libsass with Dart Sass as our Sass compiler given Libsass was deprecated.
- Migrated from Jekyll to Hugo for building our documentation

Browser support

- Dropped Internet Explorer 10 and 11
- Dropped Microsoft Edge < 16 (Legacy Edge)
- Dropped Firefox < 60
- Dropped Safari < 12
- Dropped iOS Safari < 12
- Dropped Chrome < 60

Documentation changes

- Redesigned homepage, docs layout, and footer.
- Added new Parcel guide.
- Added new Customize section, replacing v4's Theming page, with new details on Sass, global configuration options, color schemes, CSS variables, and more.
- Reorganized all form documentation into new Forms section, breaking apart the content into more focused pages.
- Similarly, updated the Layout section, to flesh out grid content more clearly.
- Renamed “Navs” component page to “Navs & Tabs”.
- Renamed “Checks” page to “Checks & radios”.
- Redesigned the navbar and added a new subnav to make it easier to get around our sites and docs versions.
- Added new keyboard shortcut for the search field: Ctrl + /.

Sass

- We've ditched the default Sass map merges to make it easier to remove redundant values. Keep in mind you now have to define all values in the Sass maps like `$theme-colors`. Check out how to deal with [Sass maps]({{< docsref “/customize/sass#maps-and-loops” >}}).
- Breaking Renamed `color-yiq()` function and related variables to `color-contrast()` as it's no longer related to YIQ colorspace. See #30168.
 - `$yiq-contrasted-threshold` is renamed to `$min-contrast-ratio`.
 - `$yiq-text-dark` and `$yiq-text-light` are respectively renamed to `$color-contrast-dark` and `$color-contrast-light`.

- Breaking Media query mixins parameters have changed for a more logical approach.
 - `media-breakpoint-down()` uses the breakpoint itself instead of the next breakpoint (e.g., `media-breakpoint-down(lg)` instead of `media-breakpoint-down(md)` targets viewports smaller than `lg`).
 - Similarly, the second parameter in `media-breakpoint-between()` also uses the breakpoint itself instead of the next breakpoint (e.g., `media-between(sm, lg)` instead of `media-breakpoint-between(sm, md)` targets viewports between `sm` and `lg`).
- Breaking Removed print styles and `$enable-print-styles` variable. Print display classes are still around. See #28339.
- Breaking Dropped `color()`, `theme-color()`, and `gray()` functions in favor of variables. See #29083.
- Breaking Renamed `theme-color-level()` function to `color-level()` and now accepts any color you want instead of only `$theme-color` colors. See #29083 **Watch out:** `color-level()` was later on dropped in v5.0.0-alpha3.
- Breaking Renamed `$enable-prefers-reduced-motion-media-query` and `$enable-pointer-cursor-for-buttons` to `$enable-reduced-motion` and `$enable-button-pointers` for brevity.
- Breaking Removed the `bg-gradient-variant()` mixin. Use the `.bg-gradient` class to add gradients to elements instead of the generated `.bg-gradient-*` classes.
- Breaking **Removed previously deprecated mixins:**
 - `hover`, `hover-focus`, `plain-hover-focus`, and `hover-focus-active`
 - `float()`
 - `form-control-mixin()`
 - `nav-divider()`
 - `retina-img()`
 - `text-hide()` (also dropped the associated utility class, `.text-hide`)
 - `visibility()`
 - `form-control-focus()`
- Breaking Renamed `scale-color()` function to `shift-color()` to avoid collision with Sass's own color scaling function.
- `box-shadow` mixins now allow `null` values and drop `none` from multiple arguments. See #30394.
- The `border-radius()` mixin now has a default value.

Color system

- The color system which worked with `color-level()` and `$theme-color-interval` was removed in favor of a new color system. All `lighten()` and `darken()` functions in our codebase are replaced by `tint-color()` and `shade-color()`. These functions will mix the color with either white or black instead of changing its lightness by a fixed amount. The `shift-color()` will either tint or shade a color depending on whether its weight parameter is positive or negative. See #30622 for more details.
- Added new tints and shades for every color, providing nine separate colors for each base color, as new Sass variables.
- Improved color contrast. Bumped color contrast ratio from 3:1 to 4.5:1 and updated blue, green, cyan, and pink colors to ensure WCAG 2.1 AA contrast. Also changed our color contrast color from `$gray-900` to `$black`.
- To support our color system, we've added new custom `tint-color()` and `shade-color()` functions to mix our colors appropriately.

Grid updates

- **New breakpoint!** Added new `xxl` breakpoint for 1400px and up. No changes to all other breakpoints.
- **Improved gutters.** Gutters are now set in rems, and are narrower than v4 (1.5rem, or about 24px, down from 30px). This aligns our grid system's gutters with our spacing utilities.
 - Added new gutter class (`.g-*`, `.gx-*`, and `.gy-*`) to control horizontal/vertical gutters, horizontal gutters, and vertical gutters.
 - Breaking Renamed `.no-gutters` to `.g-0` to match new gutter utilities.
- Columns no longer have `position: relative` applied, so you may have to add `.position-relative` to some elements to restore that behavior.
- Breaking Dropped several `.order-*` classes that often went unused. We now only provide `.order-1` to `.order-5` out of the box.
- Breaking Dropped the `.media` component as it can be easily replicated with utilities. See #28265 and the [flex utilities page for an example]({{< docsref "/utilities/flex#media-object" >}}).
- Breaking `bootstrap-grid.css` now only applies `box-sizing: border-box` to the column instead of resetting the global box-sizing. This way, our grid styles can be used in more places without interference.
- `$enable-grid-classes` no longer disables the generation of container classes anymore. See #29146.

- Updated the `make-col` mixin to default to equal columns without a specified size.

Content, Reboot, etc

- `[RFS]({{< docsref "/getting-started/rfs" >}})` is now enabled by default. Headings using the `font-size()` mixin will automatically adjust their `font-size` to scale with the viewport. *This feature was previously opt-in with v4.*
- Breaking Overhauled our display typography to replace our `$display-*` variables and with a `$display-font-sizes` Sass map. Also removed the individual `$display-*-weight` variables for a single `$display-font-weight` and adjusted `font-sizes`.
- Added two new `.display-*` heading sizes, `.display-5` and `.display-6`.
- **Links are underlined by default** (not just on hover), unless they're part of specific components.
- **Redesigned tables** to refresh their styles and rebuild them with CSS variables for more control over styling.
- Breaking Nested tables do not inherit styles anymore.
- Breaking `.thead-light` and `.thead-dark` are dropped in favor of the `.table-*` variant classes which can be used for all table elements (`thead`, `tbody`, `tfoot`, `tr`, `th` and `td`).
- Breaking The `table-row-variant()` mixin is renamed to `table-variant()` and accepts only 2 parameters: `$color` (color name) and `$value` (color code). The border color and accent colors are automatically calculated based on the table factor variables.
- Split table cell padding variables into `-y` and `-x`.
- Breaking Dropped `.pre-scrollable` class. See #29135
- Breaking `.text-*` utilities do not add hover and focus states to links anymore. `.link-*` helper classes can be used instead. See #29267
- Breaking Dropped `.text-justify` class. See #29793
- Breaking `<hr>` elements now use `height` instead of `border` to better support the `size` attribute. This also enables use of padding utilities to create thicker dividers (e.g., `<hr class="py-1">`).
- Reset default horizontal `padding-left` on `` and `` elements from browser default 40px to 2rem.
- Added `$enable-smooth-scroll`, which applies `scroll-behavior: smooth` globally—except for users asking for reduced motion through `prefers-reduced-motion` media query. See #31877

RTL

- Horizontal direction specific variables, utilities, and mixins have all been renamed to use logical properties like those found in flexbox layouts—e.g., `start` and `end` in lieu of `left` and `right`.

Forms

- **Added new floating forms!** We've promoted the Floating labels example to fully supported form components. [See the new Floating labels page.](</forms/floating-labels>)
- **Breaking Consolidated native and custom form elements.** Checkboxes, radios, selects, and other inputs that had native and custom classes in v4 have been consolidated. Now nearly all our form elements are entirely custom, most without the need for custom HTML.
 - `.custom-check` is now `.form-check`.
 - `.custom-check.custom-switch` is now `.form-check.form-switch`.
 - `.custom-select` is now `.form-select`.
 - `.custom-file` and `.form-file` have been replaced by custom styles on top of `.form-control`.
 - `.custom-range` is now `.form-range`.
 - Dropped native `.form-control-file` and `.form-control-range`.
- **Breaking Dropped `.input-group-append` and `.input-group-prepend`.** You can now just add buttons and `.input-group-text` as direct children of the input groups.
- The longstanding Missing border radius on input group with validation feedback bug is finally fixed by adding an additional `.has-validation` class to input groups with validation.
- **Breaking Dropped form-specific layout classes for our grid system.** Use our grid and utilities instead of `.form-group`, `.form-row`, or `.form-inline`.
- **Breaking Form labels now require `.form-label`.**
- **Breaking `.form-text` no longer sets `display`,** allowing you to create inline or block help text as you wish just by changing the HTML element.
- Validation icons are no longer applied to `<select>`s with `multiple`.
- Rearranged source Sass files under `scss/forms/`, including input group styles.

Components

- Unified `padding` values for alerts, breadcrumbs, cards, dropdowns, list groups, modals, popovers, and tooltips to be based on our `$spacer` variable.

See #30564.

Accordion

- Added [new accordion component]({{< docsref “/components/accordion” >}}).

Alerts

- Alerts now have [examples with icons]({{< docsref “/components/alerts#icons” >}}).
- Removed custom styles for `<hr>`s in each alert since they already use `currentColor`.

Badges

- Breaking Dropped all `.badge-*` color classes for background utilities (e.g., use `.bg-primary` instead of `.badge-primary`).
- Breaking Dropped `.badge-pill`—use the `.rounded-pill` utility instead.
- Breaking Removed hover and focus styles for `<a>` and `<button>` elements.
- Increased default padding for badges from `.25em/.5em` to `.35em/.65em`.

Breadcrumbs

- Simplified the default appearance of breadcrumbs by removing `padding`, `background-color`, and `border-radius`.
- Added new CSS custom property `--bs-breadcrumb-divider` for easy customization without needing to recompile CSS.

Buttons

- Breaking **Toggle buttons, with checkboxes or radios, no longer require JavaScript and have new markup.** We no longer require a wrapping element, add `.btn-check` to the `<input>`, and pair it with any `.btn` classes on the `<label>`. See #30650. *The docs for this has moved from our Buttons page to the new Forms section.*
- Breaking **Dropped `.btn-block` for utilities.** Instead of using `.btn-block` on the `.btn`, wrap your buttons with `.d-grid` and a `.gap-*` utility to space them as needed. Switch to responsive classes for even more control over them. Read the docs for some examples.
- Updated our `button-variant()` and `button-outline-variant()` mixins to support additional parameters.
- Updated buttons to ensure increased contrast on hover and active states.

- Disabled buttons now have `pointer-events: none;`.

Card

- Breaking Dropped `.card-deck` in favor of our grid. Wrap your cards in column classes and add a parent `.row-cols-*` container to recreate card decks (but with more control over responsive alignment).
- Breaking Dropped `.card-columns` in favor of Masonry. See #28922.
- Breaking Replaced the `.card` based accordion with a [new Accordion component]({{< docsref “/components/accordion” >}}).

Carousel

- Added new [`.carousel-dark` variant]({{< docsref “/components/carousel#dark-variant” >}}) for dark text, controls, and indicators (great for lighter backgrounds).
- Replaced chevron icons for carousel controls with new SVGs from [Bootstrap Icons]({{< param “icons” >}}).

Close button

- Breaking Renamed `.close` to `.btn-close` for a less generic name.
- Close buttons now use a `background-image` (embedded SVG) instead of a `×` in the HTML, allowing for easier customization without the need to touch your markup.
- Added new `.btn-close-white` variant that uses `filter: invert(1)` to enable higher contrast dismiss icons against darker backgrounds.

Collapse

- Removed scroll anchoring for accordions.

Dropdowns

- Added new `.dropdown-menu-dark` variant and associated variables for on-demand dark dropdowns.
- Added new variable for `$dropdown-padding-x`.
- Darkened the dropdown divider for improved contrast.
- Breaking All the events for the dropdown are now triggered on the dropdown toggle button and then bubbled up to the parent element.
- Dropdown menus now have a `data-bs-popover="static"` attribute set when the positioning of the dropdown is static and `data-bs-popover="none"` when dropdown is in the navbar. This is added by our JavaScript and

helps us use custom position styles without interfering with Popper's positioning.

- Breaking Dropped **flip** option for dropdown plugin in favor of native Popper configuration. You can now disable the flipping behavior by passing an empty array for **fallbackPlacements** option in flip modifier.
- Dropdown menus can now be clickable with a new **autoClose** option to handle the [auto close behavior]({{< docsref "/components/dropdowns#auto-close-behavior" >}}). You can use this option to accept the click inside or outside the dropdown menu to make it interactive.
- Dropdowns now support **.dropdown-items** wrapped in ****s.

Jumbotron

- Breaking Dropped the jumbotron component as it can be replicated with utilities. See our new Jumbotron example for a demo.

List group

- Added new [**.list-group-numbered** modifier]({{< docsref "/components/list-group#numbered" >}}) to list groups.

Navs and tabs

- Added new **null** variables for **font-size**, **font-weight**, **color**, and **:hover color** to the **.nav-link** class.

Navbars

- Breaking Navbars now require a container within (to drastically simplify spacing requirements and CSS required).

Offcanvas

- Added the new [offcanvas component]({{< docsref "/components/offcanvas" >}}).

Pagination

- Pagination links now have customizable **margin-left** that are dynamically rounded on all corners when separated from one another.
- Added **transitions** to pagination links.

Popovers

- Breaking Renamed **.arrow** to **.popover-arrow** in our default popover template.

- Renamed `whiteList` option to `allowList`.

Spinners

- Spinners now honor `prefers-reduced-motion`: `reduce` by slowing down animations. See #31882.
- Improved spinner vertical alignment.

Toasts

- Toasts can now be `[positioned]({{< docsref “/components/toasts#placement” >}})` in a `.toast-container` with the help of `[positioning utilities]({{< docsref “/utilities/position” >}})`.
- Changed default toast duration to 5 seconds.
- Removed `overflow: hidden` from toasts and replaced with proper `border-radius` with `calc()` functions.

Tooltips

- Breaking Renamed `.arrow` to `.tooltip-arrow` in our default tooltip template.
- Breaking The default value for the `fallbackPlacements` is changed to `['top', 'right', 'bottom', 'left']` for better placement of popper elements.
- Breaking Renamed `whiteList` option to `allowList`.

Utilities

- Breaking Renamed several utilities to use logical property names instead of directional names with the addition of RTL support:
 - Renamed `.left-*` and `.right-*` to `.start-*` and `.end-*`.
 - Renamed `.float-left` and `.float-right` to `.float-start` and `.float-end`.
 - Renamed `.border-left` and `.border-right` to `.border-start` and `.border-end`.
 - Renamed `.rounded-left` and `.rounded-right` to `.rounded-start` and `.rounded-end`.
 - Renamed `.ml-*` and `.mr-*` to `.ms-*` and `.me-*`.
 - Renamed `.pl-*` and `.pr-*` to `.ps-*` and `.pe-*`.
 - Renamed `.text-left` and `.text-right` to `.text-start` and `.text-end`.
- Breaking Disabled negative margins by default.

- Added new `.bg-body` class for quickly setting the `<body>`'s background to additional elements.
- Added new [position utilities]({{< docsref “/utilities/position#arrange-elements” >}}) for `top`, `right`, `bottom`, and `left`. Values include 0, 50%, and 100% for each property.
- Added new `.translate-middle-x` & `.translate-middle-y` utilities to horizontally or vertically center absolute/fixed positioned elements.
- Added new [border-width utilities]({{< docsref “/utilities/borders#border-width” >}}).
- Breaking Renamed `.text-monospace` to `.font-monospace`.
- Breaking Removed `.text-hide` as it's an antiquated method for hiding text that shouldn't be used anymore.
- Added `.fs-*` utilities for `font-size` utilities (with RFS enabled). These use the same scale as HTML's default headings (1-6, large to small), and can be modified via Sass map.
- Breaking Renamed `.font-weight-*` utilities as `.fw-*` for brevity and consistency.
- Breaking Renamed `.font-style-*` utilities as `.fst-*` for brevity and consistency.
- Added `.d-grid` to display utilities and new `gap` utilities (`.gap`) for CSS Grid and flexbox layouts.
- Breaking Removed `.rounded-sm` and `rounded-lg`, and introduced a new scale of classes, `.rounded-0` to `.rounded-3`. See #31687.
- Added new `line-height` utilities: `.lh-1`, `.lh-sm`, `.lh-base` and `.lh-lg`. See [here]({{< docsref “/utilities/text#line-height” >}}).
- Moved the `.d-none` utility in our CSS to give it more weight over other display utilities.
- Extended the `.visually-hidden-focusable` helper to also work on containers, using `:focus-within`.

Helpers

- Breaking **Responsive embed helpers have been renamed to [ratio helpers]**({{< docsref “/helpers/ratio” >}}) with new class names and improved behaviors, as well as a helpful CSS variable.
 - Classes have been renamed to change by to `x` in the aspect ratio. For example, `.ratio-16by9` is now `.ratio-16x9`.

- We’ve dropped the `.embed-responsive-item` and element group selector in favor of a simpler `.ratio > *` selector. No more class is needed, and the ratio helper now works with any HTML element.
- The `$embed-responsive-aspect-ratios` Sass map has been renamed to `$aspect-ratios` and its values have been simplified to include the class name and the percentage as the **key: value** pair.
- CSS variables are now generated and included for each value in the Sass map. Modify the `--bs-aspect-ratio` variable on the `.ratio` to create any [custom aspect ratio]({{< docsref “/helpers/ratio#custom-ratios” >}}).
- Breaking **“Screen reader” classes are now [“visually hidden” classes]**({{< docsref “/helpers/visually-hidden” >}}).
 - Changed the Sass file from `scss/helpers/_screenreaders.scss` to `scss/helpers/_visually-hidden.scss`
 - Renamed `.sr-only` and `.sr-only-focusable` to `.visually-hidden` and `.visually-hidden-focusable`
 - Renamed `sr-only()` and `sr-only-focusable()` mixins to `visually-hidden()` and `visually-hidden-focusable()`.
- `bootstrap-utilities.css` now also includes our helpers. Helpers don’t need to be imported in custom builds anymore.

JavaScript

- **Dropped jQuery dependency** and rewrote plugins to be in regular JavaScript.
- Breaking Data attributes for all JavaScript plugins are now namespaced to help distinguish Bootstrap functionality from third parties and your own code. For example, we use `data-bs-toggle` instead of `data-toggle`.
- **All plugins can now accept a CSS selector as the first argument.** You can either pass a DOM element or any valid CSS selector to create a new instance of the plugin:


```
var modal = new bootstrap.Modal('#myModal')
var dropdown = new bootstrap.Dropdown('[data-bs-toggle="dropdown"]')
```
- `popperConfig` can be passed as a function that accepts the Bootstrap’s default Popper config as an argument, so that you can merge this default configuration in your way. **Applies to dropdowns, popovers, and tooltips.**
- The default value for the `fallbackPlacements` is changed to `['top', 'right', 'bottom', 'left']` for better placement of Popper elements. **Applies to dropdowns, popovers, and tooltips.**
- Removed underscore from public static methods like `_getInstance()` → `getInstance()`.