

Ctrl + /

v5.1 ▾

[View on GitHub](#)

Navbar

Documentation and examples for Bootstrap's powerful, responsive navigation header, the navbar. Includes support for branding, navigation, and more, including support for our collapse plugin.



Help us build the next hub for your web development journey. Claim your Domain.

ads via Carbon

On this page

How it works

Supported content

Brand

Text

Image

Image and text

Nav

Forms

Text

Color schemes

Containers

Placement

Scrolling

Responsive behaviors

Toggler

External content

Offcanvas

Sass

Variables

Loop

How it works

Here's what you need to know before getting started with the navbar:

- Navbars require a wrapping `.navbar` with `.navbar-expand{-sm|-md|-lg|-xl|-xxl}` for responsive collapsing and [color scheme](#) classes.
- Navbars and their contents are fluid by default. Change the [container](#) to limit their horizontal width in different ways.
- Use our [spacing](#) and [flex](#) utility classes for controlling spacing and alignment within navbars.
- Navbars are responsive by default, but you can easily modify them to change that. Responsive behavior depends on our Collapse JavaScript plugin.
- Ensure accessibility by using a `<nav>` element or, if using a more generic element such as a `<div>`, add a `role="navigation"` to every navbar to explicitly identify it as a landmark region for users of assistive technologies.
- Indicate the current item by using `aria-current="page"` for the current page or `aria-current="true"` for the current item in a set.

The animation effect of this component is dependent on the `prefers-reduced-motion` media query. See the [reduced motion section of our accessibility documentation](#).

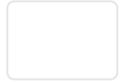
Supported content

Navbars come with built-in support for a handful of sub-components. Choose from the following as needed:

- `.navbar-brand` for your company, product, or project name.
- `.navbar-nav` for a full-height and lightweight navigation (including support for dropdowns).
- `.navbar-toggler` for use with our collapse plugin and other [navigation toggling](#) behaviors.
- Flex and spacing utilities for any form controls and actions.
- `.navbar-text` for adding vertically centered strings of text.
- `.collapse.navbar-collapse` for grouping and hiding navbar contents by a parent breakpoint.
- Add an optional `.navbar-scroll` to set a `max-height` and [scroll expanded navbar content](#).

Here's an example of all the sub-components included in a responsive light-themed navbar that automatically collapses at the **lg** (large) breakpoint.

Navbar



```
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#"
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Link</a>
        </li>
        <li class="nav-item dropdown">
          <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-
            Dropdown
          </a>
          <ul class="dropdown-menu" aria-labelledby="navbarDropdown">
            <li><a class="dropdown-item" href="#">Action</a></li>
            <li><a class="dropdown-item" href="#">Another action</a></li>
            <li><hr class="dropdown-divider"></li>
            <li><a class="dropdown-item" href="#">Something else here</a></li>
          </ul>
        </li>
        <li class="nav-item">
          <a class="nav-link disabled">Disabled</a>
        </li>
      </ul>
      <form class="d-flex">
        <input class="form-control me-2" type="search" placeholder="Search" aria-label="Sear
        <button class="btn btn-outline-success" type="submit">Search</button>
      </form>
    </div>
  </div>
</nav>
```

This example uses [background](#) (`bg-light`) and [spacing](#) (`my-2`, `my-lg-0`, `me-sm-0`, `my-sm-0`) utility classes.

Brand

The `.navbar-brand` can be applied to most elements, but an anchor works best, as some elements might require utility classes or custom styles.

Text

Add your text within an element with the `.navbar-brand` class.

Navbar

Navbar

```
<!-- As a link -->
<nav class="navbar navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
  </div>
</nav>

<!-- As a heading -->
<nav class="navbar navbar-light bg-light">
  <div class="container-fluid">
    <span class="navbar-brand mb-0 h1">Navbar</span>
  </div>
</nav>
```

Image

You can replace the text within the `.navbar-brand` with an ``.



```
<nav class="navbar navbar-light bg-light">
  <div class="container">
```

```

<a class="navbar-brand" href="#">
  
</a>
</div>
</nav>

```

Image and text

You can also make use of some additional utilities to add an image and text at the same time. Note the addition of `.d-inline-block` and `.align-text-top` on the ``.



```

<nav class="navbar navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">
      
  </div>
</nav>

```

Nav

Navbar navigation links build on our `.nav` options with their own modifier class and require the use of [toggler classes](#) for proper responsive styling. **Navigation in navbars will also grow to occupy as much horizontal space as possible** to keep your navbar contents securely aligned.

Add the `.active` class on `.nav-link` to indicate the current page.

Please note that you should also add the `aria-current` attribute on the active `.nav-link`.

Navbar



```

<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">

```

```

<a class="navbar-brand" href="#">Navbar</a>
<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#"
  <span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbarNav">
  <ul class="navbar-nav">
    <li class="nav-item">
      <a class="nav-link active" aria-current="page" href="#">Home</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Features</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Pricing</a>
    </li>
    <li class="nav-item">
      <a class="nav-link disabled">Disabled</a>
    </li>
  </ul>
</div>
</nav>

```



And because we use classes for our navs, you can avoid the list-based approach entirely if you like.

Navbar



```

<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#"
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
      <div class="navbar-nav">
        <a class="nav-link active" aria-current="page" href="#">Home</a>
        <a class="nav-link" href="#">Features</a>
        <a class="nav-link" href="#">Pricing</a>
        <a class="nav-link disabled">Disabled</a>
      </div>
    </div>
  </div>

```

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

You can also use dropdowns in your navbar. Dropdown menus require a wrapping element for positioning, so be sure to use separate and nested elements for `.nav-item` and `.nav-link` as shown below.

Navbar



```
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#"
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNavDropdown">
      <ul class="navbar-nav">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Features</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Pricing</a>
        </li>
        <li class="nav-item dropdown">
          <a class="nav-link dropdown-toggle" href="#" id="navbarDropdownMenuLink" role="but
            Dropdown link
          </a>
          <ul class="dropdown-menu" aria-labelledby="navbarDropdownMenuLink">
            <li><a class="dropdown-item" href="#">Action</a></li>
            <li><a class="dropdown-item" href="#">Another action</a></li>
            <li><a class="dropdown-item" href="#">Something else here</a></li>
          </ul>
        </li>
      </ul>
    </div>
  </div>
</nav>
```

Forms

Place various form controls and components within a navbar:

```
<nav class="navbar navbar-light bg-light">
  <div class="container-fluid">
    <form class="d-flex">
      <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search" />
      <button class="btn btn-outline-success" type="submit">Search</button>
    </form>
  </div>
</nav>
```

Immediate child elements of `.navbar` use flex layout and will default to `justify-content: space-between`. Use additional [flex utilities](#) as needed to adjust this behavior.

Navbar

```
<nav class="navbar navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand">Navbar</a>
    <form class="d-flex">
      <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search" />
      <button class="btn btn-outline-success" type="submit">Search</button>
    </form>
  </div>
</nav>
```

Input groups work, too. If your navbar is an entire form, or mostly a form, you can use the `<form>` element as the container and save some HTML.

@

Username

```

<nav class="navbar navbar-light bg-light">
  <form class="container-fluid">
    <div class="input-group">
      <span class="input-group-text" id="basic-addon1">@</span>
      <input type="text" class="form-control" placeholder="Username" aria-label="Username" a
    </div>
  </form>
</nav>

```



Various buttons are supported as part of these navbar forms, too. This is also a great reminder that vertical alignment utilities can be used to align different sized elements.

Main button

Smaller button

```

<nav class="navbar navbar-light bg-light">
  <form class="container-fluid justify-content-start">
    <button class="btn btn-outline-success me-2" type="button">Main button</button>
    <button class="btn btn-sm btn-outline-secondary" type="button">Smaller button</button>
  </form>
</nav>

```

Text

Navbars may contain bits of text with the help of `.navbar-text`. This class adjusts vertical alignment and horizontal spacing for strings of text.

Navbar text with an inline element

```

<nav class="navbar navbar-light bg-light">
  <div class="container-fluid">
    <span class="navbar-text">

```

```

    Navbar text with an inline element
  </span>
</div>
</nav>

```

Mix and match with other components and utilities as needed.

Navbar w/ text



```

<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar w/ text</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#"
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarText">
      <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Features</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Pricing</a>
        </li>
      </ul>
      <span class="navbar-text">
        Navbar text with an inline element
      </span>
    </div>
  </div>
</nav>

```

Color schemes

Theming the navbar has never been easier thanks to the combination of theming classes and `background-color` utilities. Choose from `.navbar-light` for use with light background colors, or

`.navbar-dark` for dark background colors. Then, customize with `.bg-*` utilities.

Navbar

Navbar

Navbar



```
<nav class="navbar navbar-dark bg-dark">
  <!-- Navbar content -->
</nav>
```

```
<nav class="navbar navbar-dark bg-primary">
  <!-- Navbar content -->
</nav>
```

```
<nav class="navbar navbar-light" style="background-color: #e3f2fd;">
  <!-- Navbar content -->
</nav>
```

Containers

Although it's not required, you can wrap a navbar in a `.container` to center it on a page—though note that an inner container is still required. Or you can add a container inside the `.navbar` to only center the contents of a [fixed or static top navbar](#).

Navbar

```
<div class="container">
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
    <div class="container-fluid">
      <a class="navbar-brand" href="#">Navbar</a>
    </div>
  </nav>
</div>
```

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

Use any of the responsive containers to change how wide the content in your navbar is presented.

Navbar

```
<nav class="navbar navbar-expand-lg navbar-light bg-light">  
  <div class="container-md">  
    <a class="navbar-brand" href="#">Navbar</a>  
  </div>  
</nav>
```

Placement

Use our [position utilities](#) to place navbars in non-static positions. Choose from fixed to the top, fixed to the bottom, or stickied to the top (scrolls with the page until it reaches the top, then stays there). Fixed navbars use `position: fixed`, meaning they're pulled from the normal flow of the DOM and may require custom CSS (e.g., `padding-top` on the `<body>`) to prevent overlap with other elements.

Also note that `.sticky-top` uses `position: sticky`, which [isn't fully supported in every browser](#).

Default

```
<nav class="navbar navbar-light bg-light">  
  <div class="container-fluid">  
    <a class="navbar-brand" href="#">Default</a>  
  </div>  
</nav>
```

Fixed top

```
<nav class="navbar fixed-top navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Fixed top</a>
  </div>
</nav>
```

Fixed bottom

```
<nav class="navbar fixed-bottom navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Fixed bottom</a>
  </div>
</nav>
```

Sticky top

```
<nav class="navbar sticky-top navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Sticky top</a>
  </div>
</nav>
```

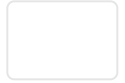
Scrolling

Add `.navbar-nav-scroll` to a `.navbar-nav` (or other navbar sub-component) to enable vertical scrolling within the toggleable contents of a collapsed navbar. By default, scrolling kicks in at `75vh` (or 75% of the viewport height), but you can override that with the local CSS custom property `--bs-navbar-height` or custom styles. At larger viewports when the navbar is expanded, content will appear as it does in a default navbar.

Please note that this behavior comes with a potential drawback of `overflow`—when setting `overflow-y: auto` (required to scroll the content here), `overflow-x` is the equivalent of `auto`, which will crop some horizontal content.

Here's an example navbar using `.navbar-nav-scroll` with `style="--bs-scroll-height: 100px;"`, with some extra margin utilities for optimum spacing.

Navbar scroll



```
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar scroll</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#"
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarScroll">
      <ul class="navbar-nav me-auto my-2 my-lg-0 navbar-nav-scroll" style="--bs-scroll-height: 100px;">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Link</a>
        </li>
        <li class="nav-item dropdown">
          <a class="nav-link dropdown-toggle" href="#" id="navbarScrollingDropdown" role="button" data-bs-toggle="dropdown">Link
          </a>
          <ul class="dropdown-menu" aria-labelledby="navbarScrollingDropdown">
            <li><a class="dropdown-item" href="#">Action</a></li>
            <li><a class="dropdown-item" href="#">Another action</a></li>
            <li><hr class="dropdown-divider"></li>
            <li><a class="dropdown-item" href="#">Something else here</a></li>
          </ul>
        </li>
        <li class="nav-item">
          <a class="nav-link disabled" href="#">Link</a>
        </li>
      </ul>
      <form class="d-flex">
        <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
        <button class="btn btn-outline-success" type="submit">Search</button>
      </form>
    </div>
  </div>
</nav>
```

Responsive behaviors

Navbars can use `.navbar-toggler`, `.navbar-collapse`, and `.navbar-expand{-sm|-md|-lg|-xl|-xxl}` classes to determine when their content collapses behind a button. In combination with other utilities, you can easily choose when to show or hide particular elements.

For navbars that never collapse, add the `.navbar-expand` class on the navbar. For navbars that always collapse, don't add any `.navbar-expand` class.

Toggler

Navbar togglers are left-aligned by default, but should they follow a sibling element like a `.navbar-brand`, they'll automatically be aligned to the far right. Reversing your markup will reverse the placement of the toggler. Below are examples of different toggle styles.

With no `.navbar-brand` shown at the smallest breakpoint:



```
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#"
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarTogglerDemo01">
      <a class="navbar-brand" href="#">Hidden brand</a>
      <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Link</a>
        </li>
        <li class="nav-item">
          <a class="nav-link disabled">Disabled</a>
        </li>
      </ul>
      <form class="d-flex">
        <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
        <button class="btn btn-outline-success" type="submit">Search</button>
      </form>
    </div>
  </div>
```

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

With a brand name shown on the left and toggler on the right:

Navbar



```
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#"
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarTogglerDemo02">
      <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Link</a>
        </li>
        <li class="nav-item">
          <a class="nav-link disabled">Disabled</a>
        </li>
      </ul>
      <form class="d-flex">
        <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
        <button class="btn btn-outline-success" type="submit">Search</button>
      </form>
    </div>
  </div>
</nav>
```

With a toggler on the left and brand name on the right:



Navbar


```

<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#"
      <span class="navbar-toggler-icon"></span>
    </button>
    <a class="navbar-brand" href="#">Navbar</a>
    <div class="collapse navbar-collapse" id="navbarTogglerDemo03">
      <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Link</a>
        </li>
        <li class="nav-item">
          <a class="nav-link disabled">Disabled</a>
        </li>
      </ul>
      <form class="d-flex">
        <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search"
        <button class="btn btn-outline-success" type="submit">Search</button>
      </form>
    </div>
  </div>
</nav>

```

External content

Sometimes you want to use the collapse plugin to trigger a container element for content that structurally sits outside of the `.navbar`. Because our plugin works on the `id` and `data-bs-target` matching, that's easily done!

```

<div class="collapse" id="navbarToggleExternalContent">
  <div class="bg-dark p-4">
    <h5 class="text-white h4">Collapsed content</h5>
    <span class="text-muted">Toggleable via the navbar brand.</span>
  </div>

```

```

</div>
<nav class="navbar navbar-dark bg-dark">
  <div class="container-fluid">
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#"
      <span class="navbar-toggler-icon"></span>
    </button>
  </div>
</nav>

```

When you do this, we recommend including additional JavaScript to move the focus programmatically to the container when it is opened. Otherwise, keyboard users and users of assistive technologies will likely have a hard time finding the newly revealed content - particularly if the container that was opened comes *before* the toggler in the document's structure. We also recommend making sure that the toggler has the `aria-controls` attribute, pointing to the `id` of the content container. In theory, this allows assistive technology users to jump directly from the toggler to the container it controls—but support for this is currently quite patchy.

Offcanvas

Transform your expanding and collapsing navbar into an offcanvas drawer with the offcanvas plugin. We extend both the offcanvas default styles and use our `.navbar-expand-*` classes to create a dynamic and flexible navigation sidebar.

In the example below, to create an offcanvas navbar that is always collapsed across all breakpoints, omit the `.navbar-expand-*` class entirely.

Offcanvas navbar



```

<nav class="navbar navbar-light bg-light fixed-top">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Offcanvas navbar</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bs-target="#"
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="offcanvas offcanvas-end" tabindex="-1" id="offcanvasNavbar" aria-labelledby="
      <div class="offcanvas-header">
        <h5 class="offcanvas-title" id="offcanvasNavbarLabel">Offcanvas</h5>
        <button type="button" class="btn-close text-reset" data-bs-dismiss="offcanvas" aria-
      </div>

```

```

<div class="offcanvas-body">
  <ul class="navbar-nav justify-content-end flex-grow-1 pe-3">
    <li class="nav-item">
      <a class="nav-link active" aria-current="page" href="#">Home</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Link</a>
    </li>
    <li class="nav-item dropdown">
      <a class="nav-link dropdown-toggle" href="#" id="offcanvasNavbarDropdown" role="
        Dropdown
      </a>
      <ul class="dropdown-menu" aria-labelledby="offcanvasNavbarDropdown">
        <li><a class="dropdown-item" href="#">Action</a></li>
        <li><a class="dropdown-item" href="#">Another action</a></li>
        <li>
          <hr class="dropdown-divider">
        </li>
        <li><a class="dropdown-item" href="#">Something else here</a></li>
      </ul>
    </li>
  </ul>
  <form class="d-flex">
    <input class="form-control me-2" type="search" placeholder="Search" aria-label="Se
    <button class="btn btn-outline-success" type="submit">Search</button>
  </form>
</div>
</div>
</div>
</nav>

```

To create an offcanvas navbar that expands into a normal navbar at a specific breakpoint like `lg`, use `.navbar-expand-lg`.

```

<nav class="navbar navbar-light navbar-expand-lg bg-light fixed-top">
  <a class="navbar-brand" href="#">Offcanvas navbar</a>
  <button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bs-target="#n
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="offcanvas offcanvas-end tabindex="-1" id="navbarOffcanvasLg" aria-labelledby=
    ...
  </div>
</nav>

```

Sass

Variables

```

$navbar-padding-y:           $spacer * .5;
$navbar-padding-x:           null;

$navbar-nav-link-padding-x:   .5rem;

$navbar-brand-font-size:      $font-size-lg;
// Compute the navbar-brand padding-y so the navbar-brand will have the same height as navba
$nav-link-height:            $font-size-base * $line-height-base + $nav-link-padding-
$navbar-brand-height:        $navbar-brand-font-size * $line-height-base;
$navbar-brand-padding-y:      ($nav-link-height - $navbar-brand-height) * .5;
$navbar-brand-margin-end:     1rem;

$navbar-toggler-padding-y:     .25rem;
$navbar-toggler-padding-x:     .75rem;
$navbar-toggler-font-size:     $font-size-lg;
$navbar-toggler-border-radius: $btn-border-radius;
$navbar-toggler-focus-width:   $btn-focus-width;
$navbar-toggler-transition:    box-shadow .15s ease-in-out;

$navbar-dark-color:           rgba($white, .55);
$navbar-dark-hover-color:     rgba($white, .75);
$navbar-dark-active-color:    $white;
$navbar-dark-disabled-color:   rgba($white, .25);
$navbar-dark-toggler-icon-bg: url("data:image/svg+xml,<svg xmlns='http://www.w3.org/20
$navbar-dark-toggler-border-color: rgba($white, .1);

$navbar-light-color:          rgba($black, .55);
$navbar-light-hover-color:    rgba($black, .7);
$navbar-light-active-color:   rgba($black, .9);
$navbar-light-disabled-color: rgba($black, .3);
$navbar-light-toggler-icon-bg: url("data:image/svg+xml,<svg xmlns='http://www.w3.org/20
$navbar-light-toggler-border-color: rgba($black, .1);

```

```

$navbar-light-brand-color: $navbar-light-active-color;
$navbar-light-brand-hover-color: $navbar-light-active-color;
$navbar-dark-brand-color: $navbar-dark-active-color;
$navbar-dark-brand-hover-color: $navbar-dark-active-color;

```

Loop

[Responsive navbar expand/collapse classes](#) (e.g., `.navbar-expand-lg`) are combined with the `$breakpoints` map and generated through a loop in `scss/_navbar.scss`.

```

// Generate series of `.navbar-expand-*` responsive classes for configuring
// where your navbar collapses.
.navbar-expand {
  @each $breakpoint in map-keys($grid-breakpoints) {
    $next: breakpoint-next($breakpoint, $grid-breakpoints);
    $infix: breakpoint-infix($next, $grid-breakpoints);

    // stylelint-disable-next-line scss/selector-no-union-class-name
    &#{$infix} {
      @include media-breakpoint-up($next) {
        flex-wrap: nowrap;
        justify-content: flex-start;

        .navbar-nav {
          flex-direction: row;

          .dropdown-menu {
            position: absolute;
          }

          .nav-link {
            padding-right: $navbar-nav-link-padding-x;
            padding-left: $navbar-nav-link-padding-x;
          }
        }

        .navbar-nav-scroll {
          overflow: visible;
        }

        .navbar-collapse {
          display: flex !important; // stylelint-disable-line declaration-no-important

```

```
    flex-basis: auto;
  }

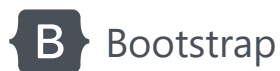
  .navbar-toggler {
    display: none;
  }

  .offcanvas-header {
    display: none;
  }

  .offcanvas {
    position: inherit;
    bottom: 0;
    z-index: 1000;
    flex-grow: 1;
    visibility: visible !important; // stylelint-disable-line declaration-no-important
    background-color: transparent;
    border-right: 0;
    border-left: 0;
    @include transition(none);
    transform: none;
  }

  .offcanvas-top,
  .offcanvas-bottom {
    height: auto;
    border-top: 0;
    border-bottom: 0;
  }

  .offcanvas-body {
    display: flex;
    flex-grow: 0;
    padding: 0;
    overflow-y: visible;
  }
}
}
}
}
```



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 v5.1.3.

Analytics by Fathom.

Links

[Home](#)

[Docs](#)

[Examples](#)

[Themes](#)

[Blog](#)

[Swag Store](#)

Projects

[Bootstrap 5](#)

[Bootstrap 4](#)

[Icons](#)

[RFS](#)

[npm starter](#)

Guides

[Getting started](#)

[Starter template](#)

[Webpack](#)

[Parcel](#)

Community

[Issues](#)

[Discussions](#)

[Corporate sponsors](#)

[Open Collective](#)

[Stack Overflow](#)