

Utilities: Adding colors

Bootstrap utility classes provide an element with styling and purpose. Let's get acquainted with utility classes by styling the color of our elements.

Let's say we want to change the text color of our paragraph element from the default black to blue. We would add a class of `"text-primary"`, like so:

```
<p class="text-primary">This text is blue!</p>
```

We could have used any of the classes provided in [Bootstrap's color example](#) to change the text color of our `<p>` element.

Styling text

If we want to style and format text, we can use [Bootstrap's text utility classes](#).

For instance, if we want our text to be bold, we could add a class of `font-weight-bold` like so:

```
<p class="font-weight-bold">  
  This rendered text is bold.  
</p>
```

We can also combine Bootstrap classes for additional styling. Let's say we wanted our text to also be uppercased. We would add `text-uppercase` to the class attribute:

```
<p class="font-weight-bold text-uppercase">  
  This rendered text is both bold and uppercased.  
</p>
```

Element Positioning

We can apply CSS positioning styling by assigning an element a utility class found in [Bootstrap's positioning documentation](#). It would also be worth knowing [how these position are rendered on a webpage](#).

The Navigation Component

Bootstrap offers a collection of *components*, such as a navbar, buttons, a carousel/slideshow for images, and much more! Each Bootstrap component serves a distinct purpose and we can find examples and code snippets directly from the documentation — then, we can tweak the components to our personal needs.

The first component we'll investigate is the *navigation (nav) component* which offers our users a collection of links.

Read through [Bootstrap's nav documentation](#) for more information.

```
<ul class="nav">
  <li class="nav-item">
    <a class="nav-link" href="#">First Link</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Second Link</a>
  </li>
</ul>
```

```
<nav class="nav">
  <a class="nav-link" href="#">First Link</a>
  <a class="nav-link" href="#">Second Link</a>
</nav>
```

The Button Component

Another common component we see on websites are buttons. By default, the `<button>` element doesn't have much styling. But, with Bootstrap we can make button components that have more styling. Here's an example from [Bootstrap's button documentation](#) :

```
<button type="button" class="btn btn-danger">Danger</button>
```

Collapsing a component

Bootstrap also allows us to quickly add interactivity to a webpage. One way to include interactivity is to toggle the visibility of an element. Look up [Bootstrap's collapse documentation](#) for additional ways of incorporating collapse.

To add such a feature, we need two elements and a few attributes — one element with content and another element that switches the visibility of the previous element. For example:

```
<button class="btn btn-primary" type="button" data-  
toggle="collapse" data-target="#collapseExample" aria-  
expanded="false" aria-controls="collapseExample">
```

This button controls the following div's visibility.
</button>

```
<div class="collapse" id="collapseExample">  
  <p>This content's visibility gets toggled</p>  
</div>
```

Creating a Navbar

We have a `<nav>` element with multiple classes:

- "navbar" makes this `<nav>` a Bootstrap navbar.
- "navbar-expand-lg" renders the `<div class="collapse navbar-collapse">` element on large (and extra large screens).
- "navbar-light" assigns a color scheme to the navbar.
- "bg-light" assigns a background color to the navbar.

Inside the `<nav>` is an `<a>` with a class of "navbar-brand" which can be an image or text that represents the brand/logo of the website.

There is a `<button>` that renders when a user's screen size is smaller than the breakpoint value in "navbar-expand-{breakpoint}" and toggles the visibility of the navbar menu to save space.

- If the user's screen size matches the breakpoint (or is bigger), then the `<div>`, with the nav component and its links, renders instead of the button.

The `` and the nested `` make up a nav component.

The first `` has a class of "active" which highlights the user's current page.

```
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <a class="navbar-brand" href="#">Brand Goes Here</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
    aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>

  <div class="collapse navbar-collapse"
    id="navbarSupportedContent">
    <ul class="navbar-nav">
      <li class="nav-item active">
        <a class="nav-link" href="#">Current Page Link
      <span class="sr-only">(current)</span></a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Another Page
        Link</a>
      </li>
    </ul>
  </div>
</nav>
```

Activity

The jumbotron component

Bootstrap offers a jumbotron component that makes content stand out. Here's an example of a basic jumbotron that only contains text:

```
<div class="jumbotron">  
  <h1>Wow this stands out!</h1>  
</div>
```

Adding a card

Bootstrap also has a *card component* that serves as a container for smaller customized content. Card components are often grouped together to display a collection of similar content in manageable chunks.

Below is an example modified from [Bootstrap's card documentation](#):

```
<div class="card">
  
  <div class="card-body">
    <h5 class="card-title">Card title</h5>
    <p class="card-text">Card description goes here.</p>
    <a href="#" class="card-link">Link to somewhere.</a>
  </div>
</div>
```

Activity

The carousel component

Bootstrap has many carousel examples but let's go through a basic example together:

The parent `<div>` element has an `id` of `"carouselExampleSlidesOnly"`, two classes `"carousel"` and `"slide"`, and the attribute `data-ride="carousel"`.

- The `id` is used later when we want to add controls to click between slides.
- The classes provide the styling and formatting.
- The `data-ride` attribute provides the interactivity and slide transitions.

We also have a nested `<div class="carousel-inner">` element that contains other `<div>` elements with images.

Nested inside the 2nd `<div>` is yet another `<div>` with a class of `"carousel-item"` and `"active"` (only one image needs the active class, if none have active, no images are shown).

Each `<div>` with `.carousel-item` has a nested `` element which have the usual `src` and `alt` attributes.

- The `` elements use two utility classes `"d-block"` sets its display as block and the `"w-100"` is to take up 100% of the width.

```
<div id="carouselExampleSlidesOnly" class="carousel
slide" data-ride="carousel">
  <div class="carousel-inner">
    <div class="carousel-item active">
      
    </div>
    <div class="carousel-item">
      
    </div>
  </div>
</div>
```

Activity