

# **UNIT-4**

# **Bootstrap**

## What is Bootstrap?

- Bootstrap is a free front-end framework for faster and easier web development
- Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins
- Bootstrap also gives you the ability to easily create responsive designs

## **What is Responsive Web Design?**

Responsive web design is about creating web sites which automatically adjust themselves to look good on all devices, from small phones to large desktops.

# Why Use Bootstrap?

## Advantages of Bootstrap:

- **Easy to use:** Anybody with just basic knowledge of HTML and CSS can start using Bootstrap
- **Responsive features:** Bootstrap's responsive CSS adjusts to phones, tablets, and desktops
- **Mobile-first approach:** In Bootstrap, mobile-first styles are part of the core framework
- **Browser compatibility:** Bootstrap 4 is compatible with all modern browsers (Chrome, Firefox, Internet Explorer 10+, Edge, Safari, and Opera)

## Where to Get Bootstrap 4?

There are two ways to start using Bootstrap 4 on your own web site.

You can:

- Include Bootstrap 4 from a CDN
- Download Bootstrap 4 from [getbootstrap.com](https://getbootstrap.com)

### Bootstrap 4 CDN

- If you don't want to download and host Bootstrap 4 yourself, you can include it from a CDN (Content Delivery Network).
- MaxCDN provides CDN support for Bootstrap's CSS and JavaScript. You must also include jQuery:

```
<!-- Latest compiled and minified CSS -->
```

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/  
bootstrap/4.4.1/css/bootstrap.min.css">
```

```
<!-- jQuery library -->
```

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/  
3.4.1/jquery.min.js"></script>
```

```
<!-- Popper JS -->
```

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/  
1.16.0/umd/popper.min.js"></script>
```

```
<!-- Latest compiled JavaScript -->
```

```
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.4.1/  
js/bootstrap.min.js"></script>
```

## **One advantage of using the Bootstrap 4 CDN:**

Many users already have downloaded Bootstrap 4 from MaxCDN when visiting another site. As a result, it will be loaded from cache when they visit your site, which leads to faster loading time. Also, most CDN's will make sure that once a user requests a file from it, it will be served from the server closest to them, which also leads to faster loading time.

## **jQuery and Popper?**

Bootstrap 4 use jQuery and Popper.js for JavaScript components (like modals, tooltips, popovers etc). However, if you just use the CSS part of Bootstrap, you don't need them.

### Show components that require jQuery »

- Closable alerts
- Buttons and checkboxes/radio buttons for toggling states
- Carousel for slides, controls, and indicators
- Collapse for toggling content
- Dropdowns (also requires Popper.js for perfect positioning)
- Modals (open and close)
- Navbar (for collapsible menus)
- Tooltips and popovers (also requires Popper.js for perfect positioning)
- Scrollspy for scroll behavior and navigation updates

# Create First Web Page With Bootstrap 4

## 1. Add the HTML5 doctype

- Bootstrap 4 uses HTML elements and CSS properties that require the HTML5 doctype.
- Always include the HTML5 doctype at the beginning of the page, along with the lang attribute and the correct character set:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
  </head>
</html>
```

## 2. Bootstrap 4 is mobile-first

- Bootstrap 4 is designed to be responsive to mobile devices. Mobile-first styles are part of the core framework.
- To ensure proper rendering and touch zooming, add the following

`<meta>` tag inside the `<head>` element:

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

- The `width=device-width` part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).
- The `initial-scale=1` part sets the initial zoom level when the page is first loaded by the browser.

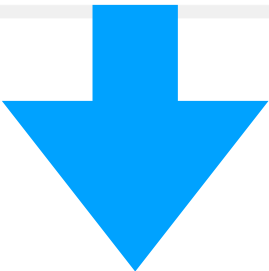
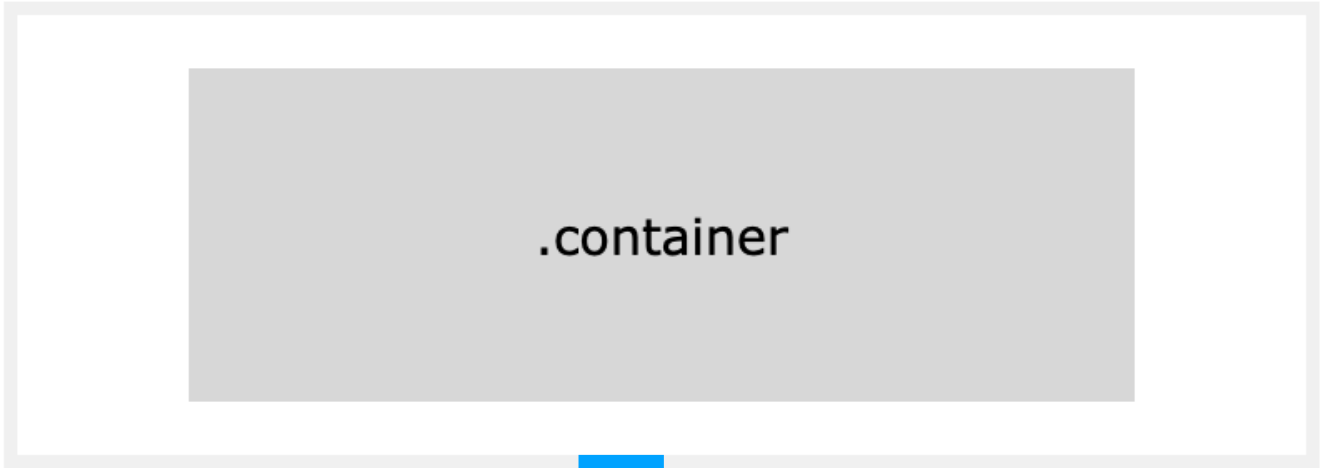
### 3. Containers

Bootstrap 4 also requires a containing element to wrap site contents.

There are two container classes to choose from:

- 1.The `.container` class provides a responsive **fixed width container**
- 2.The `.container-fluid` class provides a **full width container**, spanning the entire width of the viewport

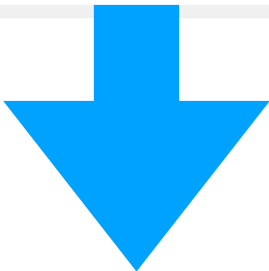
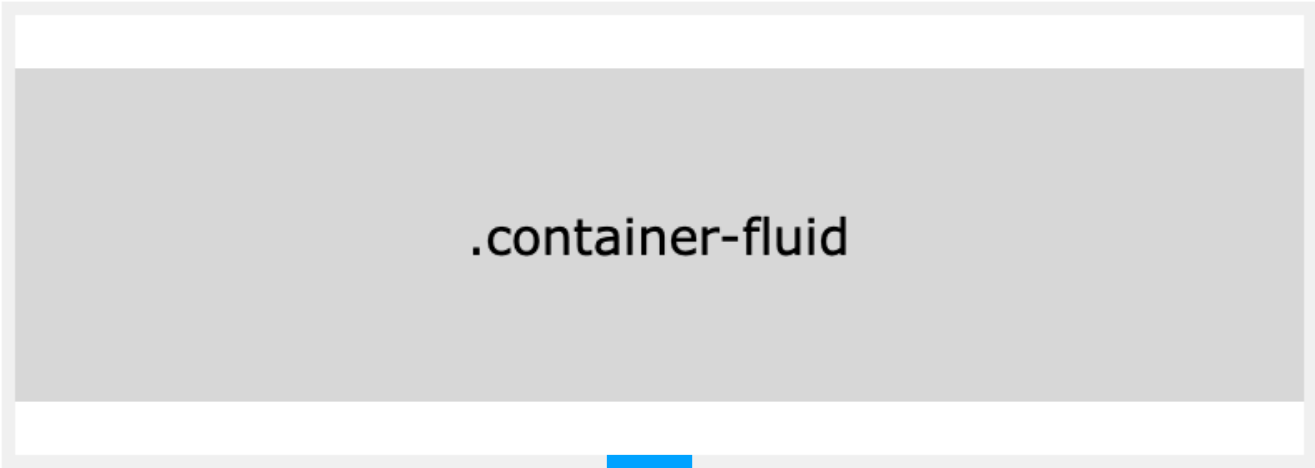




```
<body>

<div class="container">
  <h1>My First Bootstrap Page</h1>
  <p>This is some text.</p>
</div>

</body>
```



```
<body>

<div class="container-fluid">
  <h1>My First Bootstrap Page</h1>
  <p>This is some text.</p>
</div>

</body>
```

Fixed Container

Use the `.container` class to create a responsive, fixed-width container. Note that its width (`max-width`) will change on different screen sizes:

	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px
max-width	100%	540px	720px	960px	1140px

## Container Padding

- By default, containers have 15px left and right padding, with no top or bottom padding.
- Therefore, we often use **spacing utilities**, such as extra padding and margins to make them look even better.
- For example, `.pt-3` means "add a top padding of 16px":

```
<div class="container pt-3"></div>
```

## Container Border and Color

- Other utilities, such as borders and colors, are also often used together with containers:

```
<div class="container p-3 my-3 border"></div>
```

```
<div class="container p-3 my-3 bg-dark text-white"></div>
```

```
<div class="container p-3 my-3 bg-primary text-white"></div>
```

## Responsive Containers

- You can also use the `.container-sm|md|lg|xl` classes to create responsive containers.
- The `max-width` of the container will change on different screen sizes/viewports:

Class	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px
<code>.container-sm</code>	100%	540px	720px	960px	1140px
<code>.container-md</code>	100%	100%	720px	960px	1140px
<code>.container-lg</code>	100%	100%	100%	960px	1140px
<code>.container-xl</code>	100%	100%	100%	100%	1140px

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

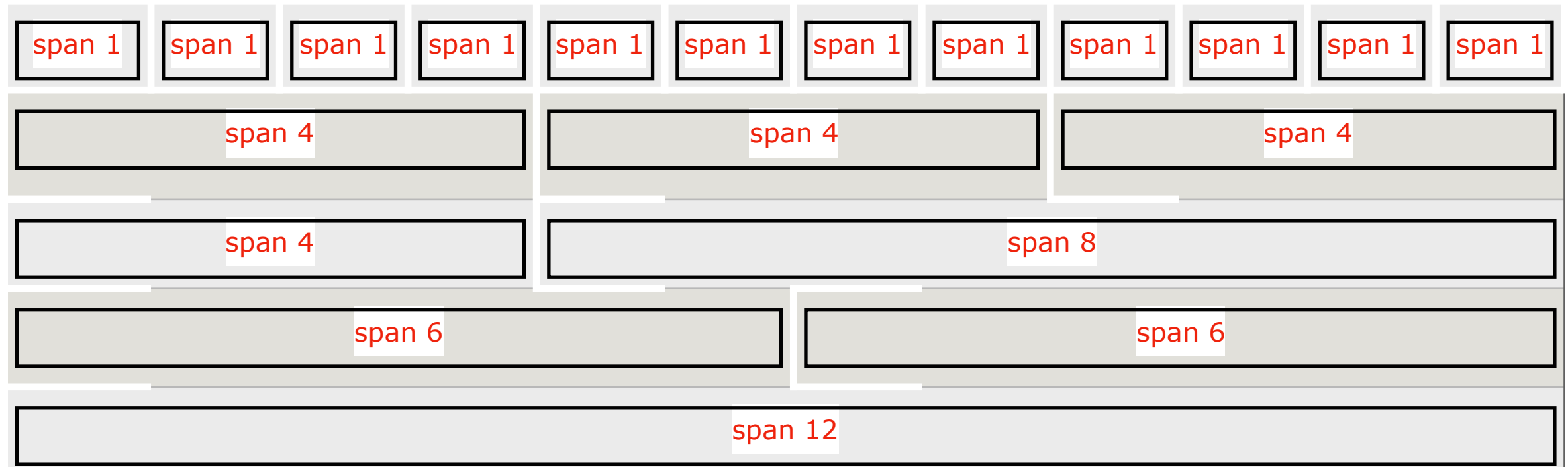
```
<div class="container-md">.container-md</div>
```

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

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

## Bootstrap 4 Grid System

- Bootstrap's grid system is built with flexbox and allows up to 12 columns across the page.
- If you do not want to use all 12 columns individually, you can group the columns together to create wider columns:



- The grid system is responsive, and the columns will re-arrange automatically depending on the screen size.
- Make sure that the sum adds up to 12 or fewer (it is not required that you use all 12 available columns).

## Grid Classes

The Bootstrap 4 grid system has five classes:

- `.col-` (extra small devices - screen width less than 576px)
- `.col-sm-` (small devices - screen width equal to or greater than 576px)
- `.col-md-` (medium devices - screen width equal to or greater than 768px)
- `.col-lg-` (large devices - screen width equal to or greater than 992px)
- `.col-xl-` (xlarge devices - screen width equal to or greater than 1200px)

The classes above can be combined to create more dynamic and flexible layouts.

**Tip:** Each class scales up, so if you wish to set the same widths for `sm` and `md`, you only need to specify `sm`.

# Basic Structure of a Bootstrap 4 Grid

```
<!-- Control the column width, and how they should appear on  
different devices -->
```

```
<div class="row">  
  <div class="col-*-*"></div>  
  <div class="col-*-*"></div>  
</div>
```

```
<div class="row">  
  <div class="col-*-*"></div>  
  <div class="col-*-*"></div>  
  <div class="col-*-*"></div>  
</div>
```

```
<!-- Or let Bootstrap automatically handle the layout -->
```

```
<div class="row">  
  <div class="col"></div>  
  <div class="col"></div>  
  <div class="col"></div>  
</div>
```

**First example:** create a row (`<div class="row">`). Then, add the desired number of columns (tags with appropriate `.col-*-*` classes).

- **The first star (\*)** represents the responsiveness: sm, md, lg or xl, while
- **The second star** represents a number, which should add up to 12 for each row.

**Second example:** instead of adding a number to each `col`, let bootstrap handle the layout, to create equal width columns:

two `"col"` elements = 50% width to each col.

three cols = 33.33% width to each col.

four cols = 25% width, etc.

You can also use `.col-sm|md|lg|xl` to make the columns responsive.

# Three Equal Columns

.col	.col	.col
------	------	------

```
<div class="row">
  <div class="col">.col</div>
  <div class="col">.col</div>
  <div class="col">.col</div>
</div>
```

# Responsive Columns

.col-sm-3	.col-sm-3	.col-sm-3	.col-sm-3
-----------	-----------	-----------	-----------

```
<div class="row">
  <div class="col-sm-3">.col-sm-3</div>
  <div class="col-sm-3">.col-sm-3</div>
  <div class="col-sm-3">.col-sm-3</div>
  <div class="col-sm-3">.col-sm-3</div>
</div>
```



# Two Unequal Responsive Columns



```
<div class="row">  
  <div class="col-sm-4">.col-sm-4</div>  
  <div class="col-sm-8">.col-sm-8</div>  
</div>
```

# Bootstrap 4 Text/Typography

## Bootstrap 4 Default Settings

- Bootstrap 4 uses a default `font-size` of 16px, and its `line-height` is 1.5.
- The default `font-family` is "Helvetica Neue", Helvetica, Arial, sans-serif.
- In addition, all `<p>` elements have `margin-top: 0` and `margin-bottom: 1rem` (16px by default).

## `<h1>` - `<h6>`

Bootstrap 4 styles HTML headings (`<h1>` to `<h6>`) with a bolder font-weight and an increased font-size:

```
<div class="container">
  <h1>h1 Bootstrap heading (2.5rem = 40px)</h1>
  <h2>h2 Bootstrap heading (2rem = 32px)</h2>
  <h3>h3 Bootstrap heading (1.75rem = 28px)</h3>
  <h4>h4 Bootstrap heading (1.5rem = 24px)</h4>
  <h5>h5 Bootstrap heading (1.25rem = 20px)</h5>
  <h6>h6 Bootstrap heading (1rem = 16px)</h6>
</div>
```

## Display Headings

Display headings are used to stand out more than normal headings (larger font-size and lighter font-weight), and there are four classes to choose from: `.display-1`, `.display-2`, `.display-3`, `.display-4`

```
<div class="container">
  <h1>Display Headings</h1>
  <p>Display headings are used to stand out more than normal
headings (larger font-size and lighter font-weight):</p>
  <h1 class="display-1">Display 1</h1>
  <h1 class="display-2">Display 2</h1>
  <h1 class="display-3">Display 3</h1>
  <h1 class="display-4">Display 4</h1>
</div>
```

### **<small>**

In Bootstrap 4 the HTML `<small>` element is used to create a lighter, secondary text in any heading:

## <mark>

Bootstrap 4 will style the HTML `<mark>` element with a yellow background color and some padding:

```
<div class="container">
  <h1>Highlight Text</h1>
  <p>Use the mark element to <mark>highlight</mark> text.</p>
</div>
```

Use the mark element to highlight text.

## <abbr>

Bootstrap 4 will style the HTML `<abbr>` element with a dotted border bottom:

## <blockquote>

Add the `.blockquote` class to a `<blockquote>` when quoting blocks of content from another source:

## <code>

Bootstrap 4 will style the HTML `<code>` element in the following way:

```
<div class="container">
  <h1>Code Snippets</h1>
  <p>Inline snippets of code should be embedded in the code
element:</p>
  <p>The following HTML elements: <code>span</code>,
<code>section</code>, and <code>div</code> defines a section in
a document.</p>
</div>
```

## Code Snippets

Inline snippets of code should be embedded in the code element:

The following HTML elements: **span**, **section**, and **div** defines a section in a document.

<kbd>

Bootstrap 4 will style the HTML **<kbd>** element in the following way:

Use **ctrl + p** to open the Print dialog box.

`<pre>`

Bootstrap 4 will style the HTML `<pre>` element in the following way:

## Bootstrap 4 Basic Table

A basic Bootstrap 4 table has a light padding and horizontal dividers. The `.table` class adds basic styling to a table:

## Striped Rows

The `.table-striped` class adds zebra-stripes to a table:

## Bordered Table

The `.table-bordered` class adds borders on all sides of the table and cells:

## Hover Rows

The `.table-hover` class adds a hover effect (grey background color) on table rows:

## Black/Dark Table

The `.table-dark` class adds a black background to the table:

## Dark Striped Table

Combine `.table-dark` and `.table-striped` to create a dark, striped table:

## Hoverable Dark Table

The `.table-hover` class adds a hover effect (grey background color) on table rows:

## Borderless Table

The `.table-borderless` class removes borders from the table:

## Contextual Classes

Contextual classes can be used to color the whole table (`<table>`), the table rows (`<tr>`) or table cells (`<td>`).

## Table Head Colors

The `.thead-dark` class adds a black background to table headers, and the `.thead-light` class adds a grey background to table headers:

## Small table

The `.table-sm` class makes the table smaller by cutting cell padding in half

## Responsive Tables

The `.table-responsive` class creates a responsive table: an horizontal scrollbar is added to the table on screens that are less than 992px wide (if needed). When viewing on anything larger than 992px wide, there is no difference:

# Buttons

Use Bootstrap's custom button styles for actions in forms, dialogs, and more with support for multiple sizes, states, and more.

## Examples

Bootstrap includes several predefined button styles, each serving its own semantic purpose, with a few extras thrown in for more control.

```
<button type="button" class="btn btn-primary">Primary</button>  
<button type="button" class="btn btn-secondary">Secondary</button>  
<button type="button" class="btn btn-success">Success</button>  
<button type="button" class="btn btn-danger">Danger</button>  
<button type="button" class="btn btn-warning">Warning</button>  
<button type="button" class="btn btn-info">Info</button>  
<button type="button" class="btn btn-light">Light</button>  
<button type="button" class="btn btn-dark">Dark</button>
```

```
<button type="button" class="btn btn-link">Link</button>
```

Primary

Secondary

Success

Danger

Warning

Info

Light

Dark

Link



## Disable text wrapping

If you don't want the button text to wrap, you can add the `.text-nowrap` class to the button. In Sass, you can set `$btn-white-space: nowrap` to disable text wrapping for each button.

## Button tags

The `.btn` classes are designed to be used with the `<button>` element. However, you can also use these classes on `<a>` or `<input>` elements (though some browsers may apply a slightly different rendering).

When using button classes on `<a>` elements that are used to trigger in-page functionality (like collapsing content), rather than linking to new pages or sections within the current page, these links should be given a `role="button"` to appropriately convey their purpose to assistive technologies such as screen readers.

```
<a class="btn btn-primary" href="#" role="button">Link</a>
<button class="btn btn-primary" type="submit">Button</button>
<input class="btn btn-primary" type="button" value="Input">
<input class="btn btn-primary" type="submit" value="Submit">
<input class="btn btn-primary" type="reset" value="Reset">
```

## Outline buttons

In need of a button, but not the hefty background colors they bring? Replace the default modifier classes with the `.btn-outline-*` ones to remove all background images and colors on any button.

```
<button type="button" class="btn btn-outline-primary">Primary</button>
<button type="button" class="btn btn-outline-secondary">Secondary</button>
<button type="button" class="btn btn-outline-success">Success</button>
<button type="button" class="btn btn-outline-danger">Danger</button>
<button type="button" class="btn btn-outline-warning">Warning</button>
<button type="button" class="btn btn-outline-info">Info</button>
<button type="button" class="btn btn-outline-light">Light</button>
<button type="button" class="btn btn-outline-dark">Dark</button>
```

## Sizes

Fancy larger or smaller buttons? Add `.btn-lg` or `.btn-sm` for additional sizes.

Create block level buttons—those that span the full width of a parent—by adding `.btn-block`.

## Active state

Buttons will appear pressed (with a darker background, darker border, and inset shadow) when active. **There's no need to add a class to `<button>`s as they use a pseudo-class.** However, you can still force the same active appearance with `.active` (and include the `aria-pressed="true"` attribute) should you need to replicate the state programmatically.

## Disabled state

Make buttons look inactive by adding the `disabled` boolean attribute to any `<button>` element.

## Toggle states

Add `data-toggle="button"` to toggle a button's `active` state. If you're pre-toggling a button, you must manually add the `.active` class **and** `aria-pressed="true"` to the `<button>`.

## Button group

Group a series of buttons together on a single line with the button group, and superpower them with JavaScript.

### Basic example

Wrap a series of buttons with `.btn` in `.btn-group`. Add on optional JavaScript radio and checkbox style behavior with [our buttons plugin](#).

```
<div class="btn-group" role="group" aria-label="Basic example">
  <button type="button" class="btn btn-secondary">Left</button>
  <button type="button" class="btn btn-secondary">Middle</button>
  <button type="button" class="btn btn-secondary">Right</button>
</div>
```