

Bootstrap 5 Get Started

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

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 5 is compatible with all modern browsers (Chrome, Firefox, Edge, Safari, and Opera). **Note** that if you need support for IE11 and down, you must use either BS4 or BS3.

Containers

Bootstrap 5 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



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	XXL ≥1400px
-----------------------	-----------------	------------------	-----------------	------------------------	----------------

max-width	100%	540px	720px	960px	1140px	1320px
-----------	------	-------	-------	-------	--------	--------

Responsive Containers

You can also use the **.container-sm|md|lg|xl** classes to determine when the container should be responsive.

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	XXL ≥1400px
.container-sm	100%	540px	720px	960px	1140px	1320px
.container-md	100%	100%	720px	960px	1140px	1320px
.container-lg	100%	100%	100%	960px	1140px	1320px
.container-xl	100%	100%	100%	100%	1140px	1320px
.container-xxl	100%	100%	100%	100%	100%	1320px

Bootstrap 5 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:

span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1
span 4				span 4				span 4			
span 4				span 8							
span 6						span 6					
span 12											

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 5 grid system has six 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)
- **.col-xxl-** (xxlarge devices - screen width equal to or greater than 1400px)

Bootstrap 5 Colors

Bootstrap 5 has some contextual classes that can be used to provide "meaning through colors".

The classes for text colors are: **.text-muted**, **.text-primary**, **.text-success**, **.text-info**, **.text-warning**, **.text-danger**, **.text-secondary**, **.text-white**, **.text-dark**, **.text-body** (default body color/often black) and **.text-light**:

Background Colors

The classes for background colors are: **.bg-primary**, **.bg-success**, **.bg-info**, **.bg-warning**, **.bg-danger**, **.bg-secondary**, **.bg-dark** and **.bg-light**.

Bootstrap 5 Tables

Basic Table

A basic Bootstrap 5 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:

Hover Rows

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

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:

Bootstrap 5 Images

Image Shapes

Rounded Corners:



Circle:



Thumbnail:



Rounded Corners

The **.rounded** class adds rounded corners to an image:

```

```

Circle

The **.rounded-circle** class shapes the image to a circle:

Aligning Images

Float an image to the left with the **.float-start** class or to the right with **.float-end**:



Example

```
  

```

Centered Image

Center an image by adding the utility classes **.mx-auto** (margin:auto) and **.d-block** (display:block) to the image:



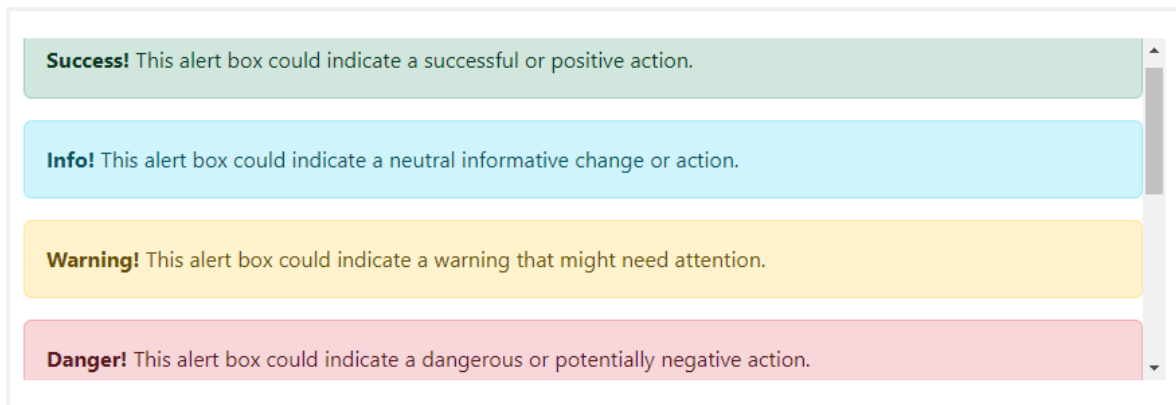
```

```

Bootstrap 5 Alerts

Alerts

Bootstrap 5 provides an easy way to create predefined alert messages:

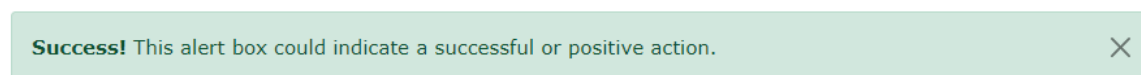


Alerts are created with the **.alert** class, followed by one of the contextual classes **.alert-success**, **.alert-info**, **.alert-warning**, **.alert-danger**, **.alert-primary**, **.alert-secondary**, **.alert-light** or **.alert-dark**

```
<div class="alert alert-success">  
  <strong>Success!</strong> Indicates a successful or  
  positive action.  
</div>
```

Closing Alerts

Closing Alerts



To close the alert message, add a `.alert-dismissible` class to the alert container. Then add `class="btn-close"` and `data-bs-dismiss="alert"` to a link or a button element (when you click on this the alert box will disappear).

```
<div class="alert alert-success alert-dismissible">  
  <button type="button" class="btn-close" data-bs-dismiss="alert"></button>  
  <strong>Success!</strong> This alert box could  
  indicate a successful or positive action.  
</div>
```

Animated Alerts

The `.fade` and `.show` classes adds a fading effect when closing the alert message:

```
<div class="alert alert-danger alert-dismissible fade  
show">
```

Bootstrap 5 Buttons

Button Styles

Bootstrap 5 provides different styles of buttons:

Basic

Primary

Secondary

Success

Info

Warning

Danger

Dark

Light

[Link](#)

```
<button type="button" class="btn">Basic</button>  
<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-  
info">Info</button>  
<button type="button" class="btn btn-  
warning">Warning</button>  
<button type="button" class="btn btn-  
danger">Danger</button>  
<button type="button" class="btn btn-  
dark">Dark</button>  
<button type="button" class="btn btn-  
light">Light</button>  
<button type="button" class="btn btn-  
link">Link</button>
```

The button classes can be used on `<a>`, `<button>`,
or `<input>` elements:

Button Outline

Bootstrap 5 also provides eight outline/bordered buttons.

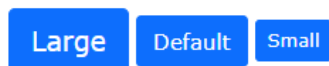
Move the mouse over them to see an additional "hover" effect:



```
<button type="button" class="btn btn-outline-primary">Primary</button>
```

Button Sizes

Use the `.btn-lg` class for large buttons or `.btn-sm` class for small buttons:



```
<button type="button" class="btn btn-primary btn-lg">Large</button>
```

Block Level Buttons

To create a block level button that spans the entire width of the parent element, use the `.d-grid` "helper" class on the parent element:

Full-Width Button

```
<div class="d-grid">  
  <button type="button" class="btn btn-primary btn-block">Full-Width Button</button>  
</div>
```

If you have many block-level buttons, you can control the space between them with the `.gap-*` class:

Full-Width Button

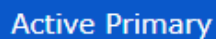
Full-Width Button

Full-Width Button

```
<div class="d-grid gap-3">  
  <button type="button" class="btn btn-primary btn-block">Full-Width Button</button>  
  <button type="button" class="btn btn-primary btn-block">Full-Width Button</button>  
  <button type="button" class="btn btn-primary btn-block">Full-Width Button</button>  
</div>
```

Active/Disabled Buttons

A button can be set to an active (appear pressed) or a disabled (unclickable) state:

A dark blue button with the text "Active Primary" in white.A light blue button with the text "Disabled Primary" in dark blue.

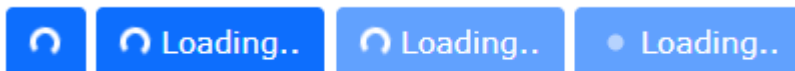
The class `.active` makes a button appear pressed, and the `disabled` attribute makes a button unclickable.

Note that `<a>` elements do not support the disabled attribute and must therefore use the `.disabled` class to make it visually appear disabled.

```
<button type="button" class="btn btn-primary  
active">Active Primary</button>  
<button type="button" class="btn btn-  
primary" disabled>Disabled Primary</button>  
<a href="#" class="btn btn-primary disabled">Disabled  
Link</a>
```

Spinner Buttons

You can also add "spinners" to a button, which you will learn more about in our [BS5 Spinners Tutorial](#):



```
<button class="btn btn-primary">  
  <span class="spinner-border spinner-border-  
sm"></span>
```

```
</button>
```

```
<button class="btn btn-primary">  
  <span class="spinner-border spinner-border-  
sm"></span>  
  Loading..  
</button>
```

Bootstrap 5 Button Groups

Button Groups

Bootstrap 5 allows you to group a series of buttons together (on a single line) in a button group:

Use a `<div>` element with class `.btn-group` to create a button group:

```
<div class="btn-group">  
  <button type="button" class="btn btn-  
primary">Apple</button>  
  <button type="button" class="btn btn-  
primary">Samsung</button>  
  <button type="button" class="btn btn-  
primary">Sony</button>  
</div>
```


Tip: Instead of applying button sizes to every button in a group, use class `.btn-group-lg` for a large button group or the `.btn-group-sm` for a small button group:

Vertical Button Groups

Use the class `.btn-group-vertical` to create a vertical button group:

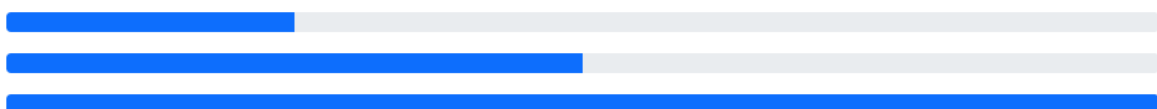
Button Dropdown Menus

```
<div class="btn-group">
  <button type="button" class="btn btn-primary
dropdown-toggle" data-bs-
toggle="dropdown">Sony</button>
  <div class="dropdown-menu">
    <a class="dropdown-item" href="#">Tablet</a>
    <a class="dropdown-
item" href="#">Smartphone</a>
  </div>
```

Bootstrap 5 Progress Bars

Basic Progress Bar

A progress bar can be used to show how far a user is in a process.



To create a default progress bar, add a `.progress` class to a container element and add the `.progress-bar` class to its child element. Use the CSS `width` property to set the width of the progress bar:

```
<div class="progress">  
  <div class="progress-bar" style="width:70%"></div>  
</div>
```

Progress Bar Height

```
<div class="progress" style="height:20px">  
  <div class="progress-bar" style="width:40%;"></div>  
</div>
```

Progress Bar Labels

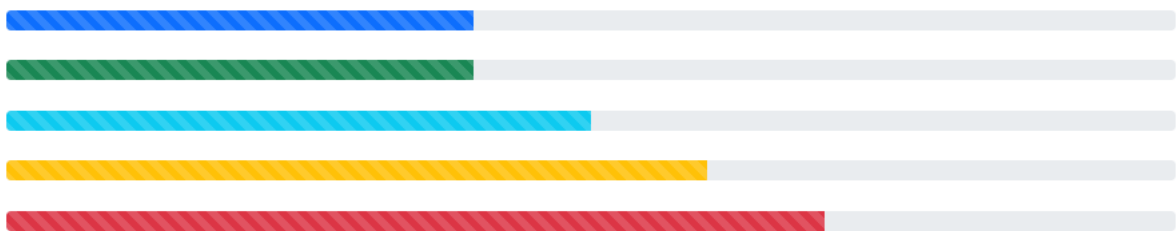
Add text inside the progress bar to show the visible percentage:

```
<div class="progress">  
  <div class="progress-  
bar" style="width:70%">70%</div>  
</div>
```

Colored Progress Bars

```
<!-- Red -->
<div class="progress">
  <div class="progress-bar bg-
danger" style="width:50%"></div>
</div>
```

Striped Progress Bars



```
<div class="progress">
  <div class="progress-bar progress-bar-
striped" style="width:40%"></div>
</div>
```

Animated Progress Bar

Add the `.progress-bar-animated` class to animate the progress bar

```
<div class="progress-bar progress-bar-striped progress-
bar-animated" style="width:40%"></div>
```

Multiple Progress Bars

Progress bars can also be stacked:



```
<div class="progress">
  <div class="progress-bar bg-
success" style="width:40%">
    Free Space
  </div>
  <div class="progress-bar bg-
warning" style="width:10%">
    Warning
  </div>
  <div class="progress-bar bg-
danger" style="width:20%">
    Danger
  </div>
</div>
```

Bootstrap 5 Spinners

Spinners

To create a spinner/loader, use the `.spinner-border` class:

```
<div class="spinner-border"></div>
```

Colored Spinners

Use any **text color utilities** to add a color to the spinner:



```
<div class="spinner-border text-danger"></div>
```

Growing Spinners

Use the **.spinner-grow** class if you want the spinner/loader to grow instead of "spin":

```
<div class="spinner-grow text-danger"></div>
```

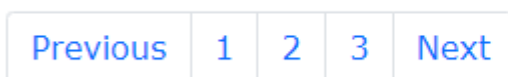
Spinner Size

Use **.spinner-border-sm** or **.spinner-grow-sm** to create a smaller spinner:

Bootstrap 5 Pagination

Basic Pagination

If you have a web site with lots of pages, you may wish to add some sort of pagination to each page.

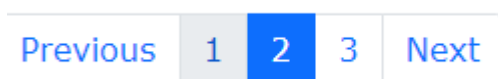


To create a basic pagination, add the **.pagination** class to an **** element. Then add the **.page-item** to each **** element and a **.page-link** class to each link inside ****:

```
<ul class="pagination">
  <li class="page-item"><a class="page-
link" href="#">Previous</a></li>
  <li class="page-item"><a class="page-
link" href="#">1</a></li>
  <li class="page-item"><a class="page-
link" href="#">2</a></li>
  <li class="page-item"><a class="page-
link" href="#">3</a></li>
  <li class="page-item"><a class="page-
link" href="#">Next</a></li>
</ul>
```

Active State

The **.active** class is used to "highlight" the current page:



```
<ul class="pagination">
  <li class="page-item"><a class="page-
link" href="#">Previous</a></li>
  <li class="page-item"><a class="page-
link" href="#">1</a></li>
  <li class="page-item active"><a class="page-
link" href="#">2</a></li>
  <li class="page-item"><a class="page-
link" href="#">3</a></li>
```

```
<li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
```

Disabled State

The `.disabled` class is used for un-clickable links:

```
<ul class="pagination">
  <li class="page-item disabled"><a class="page-link" href="#">Previous</a></li>
</ul>
```

Pagination Sizing

Add class `.pagination-lg` for larger blocks or `.pagination-sm` for smaller blocks:

```
<ul class="pagination pagination-lg">
  <li class="page-item"><a class="page-link" href="#">Previous</a></li>
</ul>
```

Breadcrumbs

Another form for pagination, is breadcrumbs:

[Photos](#) / [Summer 2017](#) / [Italy](#) / Rome

The `.breadcrumb` and `.breadcrumb-item` classes indicates the current page's location within a navigational hierarchy:

```
<ul class="breadcrumb">
  <li class="breadcrumb-
item"><a href="#">Photos</a></li>
  <li class="breadcrumb-item"><a href="#">Summer
2017</a></li>
  <li class="breadcrumb-
item"><a href="#">Italy</a></li>
  <li class="breadcrumb-item active">Rome</li>
</ul>
```

Basic List Groups

The most basic list group is an unordered list with list items:

First item
Second item
Third item

To create a basic list group, use an `` element with class `.list-group`, and `` elements with class `.list-group-item`:


```
<ul class="list-group">
  <li class="list-group-item">First item</li>
  <li class="list-group-item">Second item</li>
  <li class="list-group-item">Third item</li>
</ul>
```

Active State

Use the **.active** class to highlight the current item:

List Group With Linked Items

To create a list group with linked items, use **<div>** instead of **** and **<a>** instead of ****. Optionally, add the **.list-group-item-action** class if you want a grey background color on hover:

Disabled Item

The **.disabled** class adds a lighter text color to the disabled item. And when used on links, it will remove the hover effect:

Flush / Remove Borders

Use the **.list-group-flush** class to remove some borders and rounded corners:

```
<ul class="list-group list-group-flush">
```

Numbered List Groups

Use the `.list-group-numbered` class to create list items with numbers in front of them:

```
<ol class="list-group list-group-numbered">
```

Bootstrap 5 Dropdowns

Basic Dropdown

A dropdown menu is a toggleable menu that allows the user to choose one value from a predefined list:

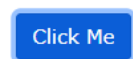
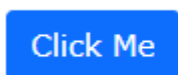
```
<div class="dropdown">
  <button type="button" class="btn btn-primary
dropdown-toggle" data-bs-toggle="dropdown">
    Dropdown button
  </button>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="#">Link
1</a></li>
    <li><a class="dropdown-item" href="#">Link
2</a></li>
    <li><a class="dropdown-item" href="#">Link
3</a></li>
  </ul>
</div>
```

The `.dropdown` class indicates a dropdown menu.

To open the dropdown menu, use a button or a link with a class of `.dropdown-toggle` and the `data-bs-toggle="dropdown"` attribute.

Basic Collapsible

Collapsibles are useful when you want to hide and show large amount of content:



Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

```
<button data-bs-toggle="collapse" data-bs-target="#demo">Collapsible</button>
```

```
<div id="demo" class="collapse">  
Lorem ipsum dolor text....  
</div>
```

Accordion

Collapsible Group Item #1
Collapsible Group Item #2
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.
Collapsible Group Item #3

Note: Use the **data-bs-parent** attribute to make sure that all collapsible elements under the specified parent will be closed when one of the collapsible item is shown.

```
<div id="accordion">
  <div class="card">
    <div class="card-header">
      <a class="btn" data-bs-
toggle="collapse" href="#collapseOne">
        Collapsible Group Item #1
      </a>
    </div>
    <div id="collapseOne" class="collapse show" data-bs-
parent="#accordion">
      <div class="card-body">
        Lorem ipsum..
      </div>
    </div>
  </div>

  <div class="card">
    <div class="card-header">
```

```
<a class="collapsed btn" data-bs-  
toggle="collapse" href="#collapseTwo">  
  Collapsible Group Item #2  
</a>  
</div>  
<div id="collapseTwo" class="collapse" data-bs-  
parent="#accordion">  
  <div class="card-body">  
    Lorem ipsum..  
  </div>  
</div>  
</div>  
  
<div class="card">  
  <div class="card-header">  
    <a class="collapsed btn" data-bs-  
toggle="collapse" href="#collapseThree">  
      Collapsible Group Item #3  
    </a>  
  </div>  
  <div id="collapseThree" class="collapse" data-bs-  
parent="#accordion">  
    <div class="card-body">  
      Lorem ipsum..  
    </div>  
  </div>  
</div>  
</div>
```

Bootstrap 5 Navs

Nav Menus

Link Link Link Disabled

If you want to create a simple horizontal menu, add the `.nav` class to a `` element, followed by `.nav-item` for each `` and add the `.nav-link` class to their links:

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

Aligned Nav

Add the `.justify-content-center` class to center the nav, and the `.justify-content-end` class to right-align the nav.

Vertical Nav

Add the `.flex-column` class to create a vertical nav:

```
<ul class="nav flex-column">
```

Tabs

Turn the nav menu into navigation tabs with the **.nav-tabs** class. Add the **.active** class to the active/current link. If you want the tabs to be togglable, see the last example on this page.

```
<ul class="nav nav-tabs">  
  <li class="nav-item">  
    <a class="nav-link active" href="#">Active</a>  
  </li>  
  <li class="nav-item">  
    <a class="nav-link disabled" href="#">Disabled</a>  
  </li>  
</ul>
```

Pills

Turn the nav menu into navigation pills with the **.nav-pills** class. If you want the pills to be togglable, see the last example on this page.

```
<ul class="nav nav-pills">
```

Tabs with Dropdown

Bootstrap 5 Navbars

Navigation Bars

A navigation bar is a navigation header that is placed at the top of the page:

Basic Navbar

With Bootstrap, a navigation bar can extend or collapse, depending on the screen size.

A standard navigation bar is created with the `.navbar` class, followed by a responsive collapsing class: `.navbar-expand-xxl|xl|lg|md|sm` (stacks the navbar vertically on xxlarge, extra large, large, medium or small screens).

To add links inside the navbar, use either an `` element (or a `<div>`) with `class="navbar-nav"`. Then add `` elements with a `.nav-item` class followed by an `<a>` element with a `.nav-link` class:

`<!-- A grey horizontal navbar that becomes vertical on small screens -->`

```
<nav class="navbar navbar-expand-sm bg-light">
```

```
<div class="container-fluid">
```

```
<!-- Links -->
```

```
<ul class="navbar-nav">
```

```
<li class="nav-item">
```

```
<a class="nav-link" href="#">Link 1</a>
```

```
</li>
```

```
<li class="nav-item">
```



```
    <a class="nav-link" href="#">Link 2</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Link 3</a>
  </li>
</ul>
</div>

</nav>
```

Vertical Navbar

Remove the `.navbar-expand-*` class to create a navigation bar that will always be vertical:

Centered Navbar

Add the `.justify-content-center` class to center the navigation bar:

Colored Navbar

Brand / Logo

The `.navbar-brand` class is used to highlight the brand/logo/project name of your page:

```
<nav class="navbar navbar-expand-sm bg-dark navbar-dark">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Logo</a>
  </div>
</nav>
```

Bootstrap 5 Slideshow

Slideshow

```
<!-- Carousel -->
<div id="demo" class="carousel slide" data-bs-ride="carousel">

  <!-- Indicators/dots -->
  <div class="carousel-indicators">
    <button type="button" data-bs-target="#demo" data-bs-slide-to="0" class="active"></button>
    <button type="button" data-bs-target="#demo" data-bs-slide-to="1"></button>
    <button type="button" data-bs-target="#demo" data-bs-slide-to="2"></button>
  </div>

  <!-- The slideshow/carousel -->
  <div class="carousel-inner">
    <div class="carousel-item active">
      
    </div>
    <div class="carousel-item">
```

```

        
    </div>
    <div class="carousel-item">
        
    </div>
</div>

<!-- Left and right controls/icons -->
<button class="carousel-control-prev" type="button"
data-bs-target="#demo" data-bs-slide="prev">
    <span class="carousel-control-prev-icon"></span>
</button>
<button class="carousel-control-next" type="button"
data-bs-target="#demo" data-bs-slide="next">
    <span class="carousel-control-next-icon"></span>
</button>
</div>

```

Example explained

A description of what each class from the example above do:

Class	Description
.carousel	Creates a carousel

**.carousel-
indicators**

Adds indicators for the carousel. These are the little dots at the bottom of each slide (which indicates how many slides there are in the carousel, and which slide the user are currently viewing)

**.carousel-
inner**

Adds slides to the carousel

**.carousel-
item**

Specifies the content of each slide

**.carousel-
control-prev**

Adds a left (previous) button to the carousel, which allows the user to go back between the slides

**.carousel-
control-next**

Adds a right (next) button to the carousel, which allows the user to go forward between the slides

**.carousel-
control-
prev-icon**

Used together with .carousel-control-prev to create a "previous" button

.carousel-control-next-icon Used together with .carousel-control-next to create a "next" button

.slide Adds a CSS transition and animation effect when sliding from one item to the next. Remove this class if you do not want this effect

Add Captions to Slides

Add elements inside `<div class="carousel-caption">` within each `<div class="carousel-item">` to create a caption for each slide:

```
<div class="carousel-item">
  
  <div class="carousel-caption">
    <h3>Los Angeles</h3>
    <p>We had such a great time in LA!</p>
  </div>
</div>
```

Bootstrap 5 Modal

Modals

The Modal component is a dialog box/popup window that is displayed on top of the current page:

```
<!-- Button to Open the Modal -->
<button type="button" class="btn btn-primary" data-bs-
toggle="modal" data-bs-target="#myModal">
  Open modal
</button>

<!-- The Modal -->
<div class="modal" id="myModal">
  <div class="modal-dialog">
    <div class="modal-content">

      <!-- Modal Header -->
      <div class="modal-header">
        <h4 class="modal-title">Modal Heading</h4>
        <button type="button" class="btn-close" data-
bs-dismiss="modal"></button>
      </div>

      <!-- Modal body -->
      <div class="modal-body">
        Modal body..
      </div>

      <!-- Modal footer -->
      <div class="modal-footer">
        <button type="button" class="btn btn-danger"
data-bs-dismiss="modal">Close</button>
      </div>

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

Add animation

Use the `.fade` class to add a fading effect when opening and closing the modal:

```
<!-- Fading modal -->  
<div class="modal fade"></div>
```

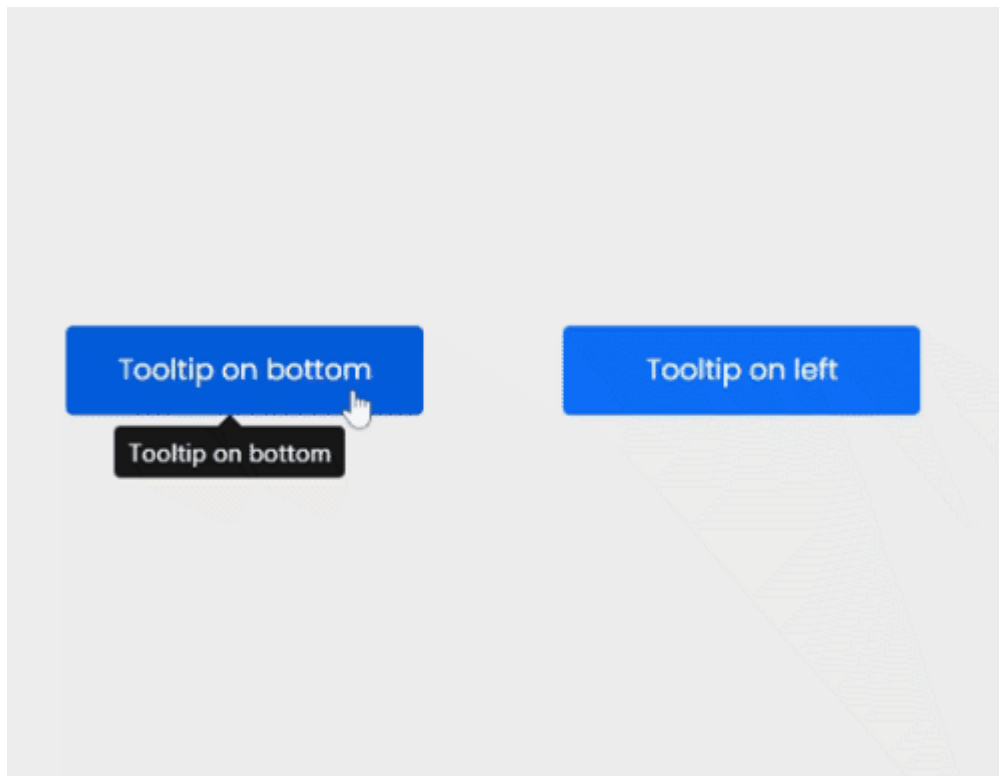
```
<!-- Modal without animation -->  
<div class="modal"></div>
```

Scrolling Modal

When you have a lot of content inside the modal, a scrollbar is added to the page. See the examples below to understand it:

```
<div class="modal-dialog">
```

Bootstrap 5 Tooltip



How To Create a Tooltip

To create a tooltip, add the `data-bs-toggle="tooltip"` attribute to an element.

Use the `title` attribute to specify the text that should be displayed inside the tooltip:

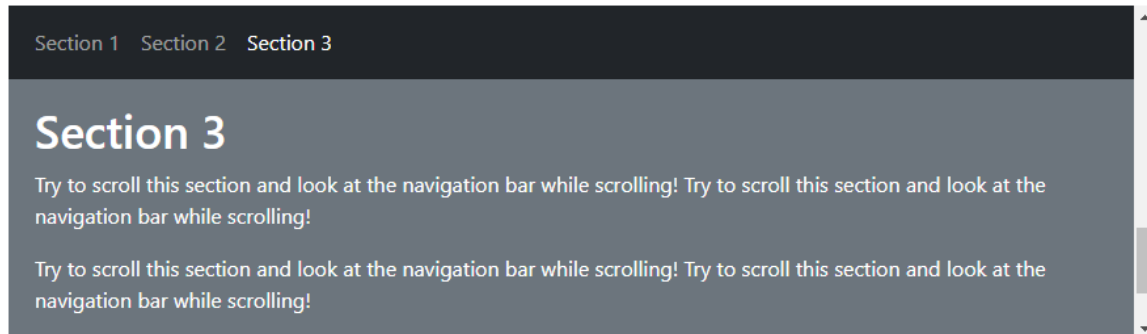
Note: Tooltips must be initialized with JavaScript to work.

```
<button type="button" class="btn btn-primary" data-bs-toggle="tooltip" title="Hooray!">Hover over me!</button>
```


Bootstrap 5 Scrollspy

Scrollspy

Scrollspy is used to automatically update links in a navigation list based on **scroll** position.



How To Create a Scrollspy

The following example shows how to create a scrollspy:

```
<!-- The scrollable area -->

<body data-bs-spy="scroll" data-bs-target=".navbar"
data-bs-offset="50">

  <!-- The navbar - The <a> elements are used to jump
to a section in the scrollable area -->
  <nav class="navbar navbar-expand-sm bg-dark
navbar-dark fixed-top">
    ...
    <ul class="navbar-nav">
      <li><a href="#section1">Section 1</a></li>
      ...
    </nav>

  <!-- Section 1 -->
  <div id="section1">
    <h1>Section 1</h1>
```

```
<p>Try to scroll this page and look at the navigation  
bar while scrolling!</p>  
</div>  
...  
</body>
```

Example Explained

Add **data-bs-spy="scroll"** to the element that should be used as the scrollable area (often this is the **<body>** element).

Then add the **data-bs-target** attribute with a value of the id or the class name of the navigation bar (**.navbar**). This is to make sure that the navbar is connected with the scrollable area.

Note that scrollable elements must match the ID of the links inside the navbar's list items (**<div id="section1">** matches ****).

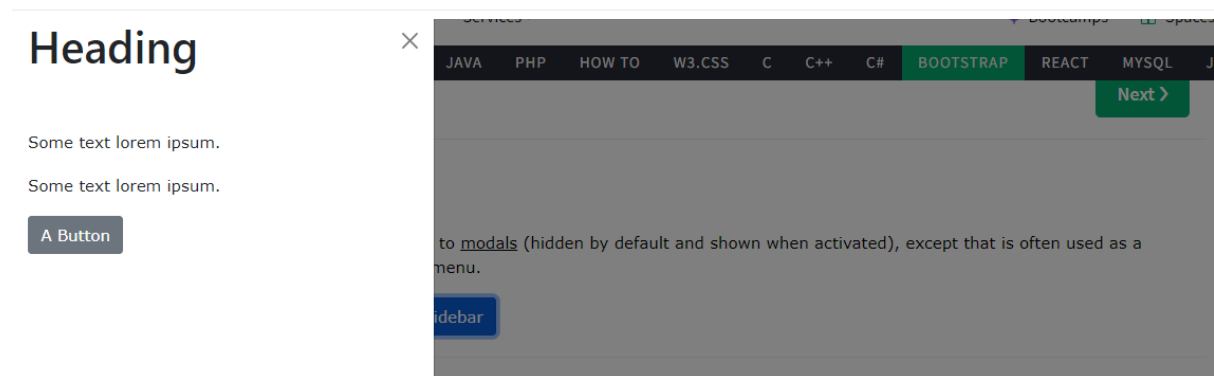
The optional **data-bs-offset** attribute specifies the number of pixels to offset from top when calculating the position of scroll

Bootstrap 5 Offcanvas

Offcanvas

Offcanvas is similar to [modals](#) (hidden by default and shown when activated), except that is often used as a sidebar navigation menu.

Open Offcanvas Sidebar



How To Create an Offcanvas Sidebar

The following example shows how to create an offcanvas sidebar:

```
<!-- Offcanvas Sidebar -->
<div class="offcanvas offcanvas-start" id="demo">
  <div class="offcanvas-header">
    <h1 class="offcanvas-title">Heading</h1>
    <button type="button" class="btn-close text-reset"
data-bs-dismiss="offcanvas"></button>
  </div>
  <div class="offcanvas-body">
    <p>Some text lorem ipsum.</p>
    <p>Some text lorem ipsum.</p>
```

```
<button class="btn btn-secondary"
type="button">A Button</button>
</div>
</div>

<!-- Button to open the offcanvas sidebar -->
<button class="btn btn-primary" type="button" data-bs-
toggle="offcanvas" data-bs-target="#demo">
  Open Offcanvas Sidebar
</button>
```

Example Explained

The **.offcanvas** class creates the offcanvas sidebar.

The **.offcanvas-start** class positions the offcanvas, and makes it 400px wide. See examples below for more positioning classes.

The **.offcanvas-title** class ensures proper margins and line-height.

Then, add your content inside the **.offcanvas-body** class.

To open the offcanvas sidebar, you must use a **<button>** or an **<a>** element that points to the id of the **.offcanvas** container (**#demo** in our example).

To open the offcanvas sidebar with an **<a>** element, you can point to **#demo** with the href attribute, instead of **data-bs-target** attribute.

Offcanvas Position

Use the **.offcanvas-start|end|top|bottom** to position the offcanvas to the left, right, top or bottom:

```
<div class="offcanvas offcanvas-end" id="demo">
```

Bootstrap 5 Utilities

Borders

Use the **border** classes to add or remove borders from an element:

```
<span class="border"></span>
```

```
<span class="border-top"></span>
```

```
<span class="border-end"></span>
```

```
<span class="border-bottom"></span>
```

```
<span class="border-start"></span>
```

Border Width

Use **.border-1** to **.border-5** to change the width of the border:

Border Radius

Add rounded corners to an element with the **rounded** classes:

```
<span class="rounded-circle"></span>
```

Float and Clearfix

Float an element to the right with the `.float-end` class or to the left with `.float-start`, and clear floats with the `.clearfix` class:

```
<div class="clearfix">  
  <span class="float-start">Float left</span>  
  <span class="float-end">Float right</span>  
</div>
```

Spacing

Where *property* is one of:

- **m** - sets **margin**
- **p** - sets **padding**

Where *sides* is one of:

- **t** - sets **margin-top** or **padding-top**
- **b** - sets **margin-bottom** or **padding-bottom**
- **s** - sets **margin-left** or **padding-left**
- **e** - sets **margin-right** or **padding-right**
- **x** - sets both **padding-left** and **padding-right** or **margin-left** and **margin-right**
- **y** - sets both **padding-top** and **padding-bottom** or **margin-top** and **margin-bottom**
- blank - sets a **margin** or **padding** on all 4 sides of the element

Vertical Align

Use the **align-** classes to change the alignment of elements (only works on inline, inline-block, inline-table and table cell elements):

```
<span class="align-baseline">baseline</span>  
<span class="align-top">top</span>  
<span class="align-middle">middle</span>  
<span class="align-bottom">bottom</span>  
<span class="align-text-top">text-top</span>  
<span class="align-text-bottom">text-bottom</span>
```

Bootstrap 5 Flex

Flexbox

The biggest difference between Bootstrap 3 and Bootstrap 4 & 5 is that Bootstrap 5 now uses flexbox, instead of floats, to handle the layout.

To create a flexbox container and to transform direct children into flex items, use the **d-flex** class:

Example



Flex item 1 Flex item 2 Flex item 3

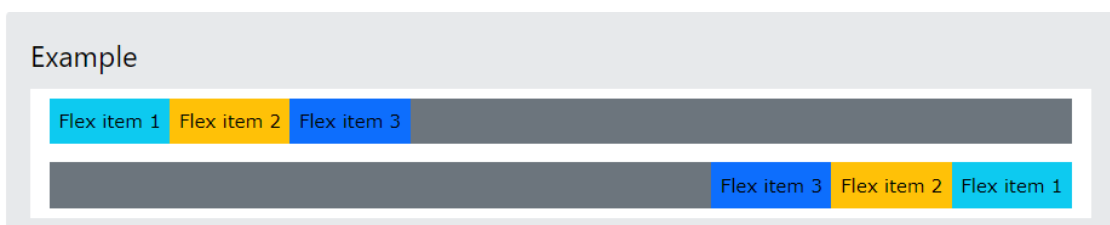
```
<div class="d-flex p-3 bg-secondary text-white">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
  <div class="p-2 bg-primary">Flex item 3</div>
</div>
```

To create an inline flexbox container, use the **d-inline-flex** class:

Horizontal Direction

Use **.flex-row** to display the flex items horizontally (side by side). This is default.

Tip: Use **.flex-row-reverse** to right-align the horizontal direction:



AD

```
<div class="d-flex flex-row bg-secondary">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
  <div class="p-2 bg-primary">Flex item 3</div>
</div>
```

```
<div class="d-flex flex-row-reverse bg-secondary">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
```



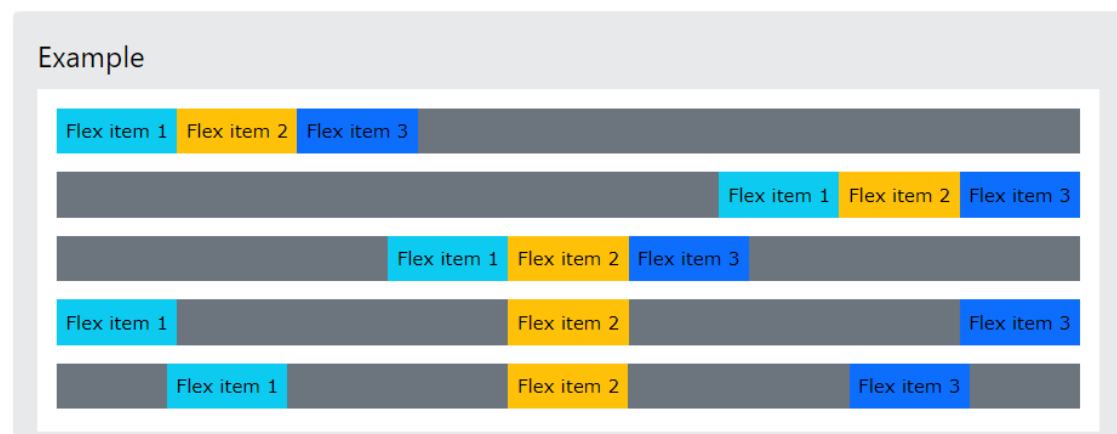
```
<div class="p-2 bg-primary">Flex item 3</div>  
</div>
```

Vertical Direction

Use **.flex-column** to display the flex items vertically (on top of each other), or **.flex-column-reverse** to reverse the vertical direction:

Justify Content

Use the **.justify-content-*** classes to change the alignment of flex items. Valid classes are **start** (default), **end**, **center**, **between** or **around**:




AD

```
<div class="d-flex justify-content-start">...</div>  
<div class="d-flex justify-content-end">...</div>  
<div class="d-flex justify-content-center">...</div>  
<div class="d-flex justify-content-between">...</div>  
<div class="d-flex justify-content-around">...</div>
```

Fill / Equal Widths

Use **.flex-fill** on flex items to force them into equal widths:

Example



Align Content

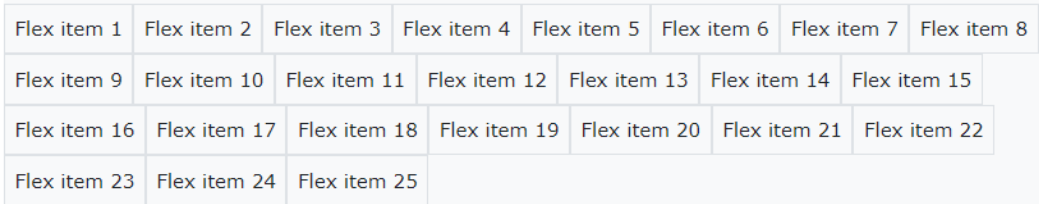
Control the vertical alignment of **gathered** flex items with the **.align-content-*** classes. Valid classes are **.align-content-start** (default), **.align-content-end**, **.align-content-center**, **.align-content-between**, **.align-content-around** and **.align-content-stretch**.

Note: These classes have no effect on single rows of flex items.

align-content-start align-content-end **align-content-center** align-content-around

align-content-stretch

Example



```
<div class="d-flex flex-wrap align-content-start">..
```

```
<div class="d-flex flex-wrap align-content-end">..
```

```
<div class="d-flex flex-wrap align-content-center">..
```

```
<div class="d-flex flex-wrap align-content-around">..
```

```
<div class="d-flex flex-wrap align-content-stretch">..
```

Align Items

Control the vertical alignment of **single rows** of flex items with the **.align-items-*** classes. Valid classes are **.align-items-start**, **.align-items-end**, **.align-items-center**, **.align-items-baseline**, and **.align-items-stretch** (default).

Bootstrap 5 Forms

Bootstrap 5 Grid System

The 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:

span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1
span 4				span 4				span 4			
span 4				span 8							
span 6						span 6					
span 12											

Grid Classes

The Bootstrap 5 grid system has six 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)
- **.col-xxl-** (xxlarge devices - screen width equal to or greater than 1400px)

List of components:

1. **Jumbotron**: It simply put extra attention to particular content or information by making it larger and more eye-catching.
2. **Alerts**: It is a popup with a predefined message that appears after a particular action.

3. **Buttons**: It is customized buttons that are used to perform an action in the form, dialogue box, etc. They are in multiple states, sizes and have predefined styles.
4. **Button group**: It is a group of buttons aligned in a single line and they can be arranged both vertically as well as horizontally.
5. **Badge**: It Is a labeling component that is used to add additional information.
6. **Progress Bar**: It is used to show the progress of a particular operation with a custom progress bar. They have text labels, stacked bars, and animated backgrounds.
7. **Spinner**: The spinner displays the loading state of websites or projects. They are built with HTML, CSS and don't require any JavaScript.
8. **Scrollspy**: It keeps updating the navigation bar to the currently active link based on the scroll position in the viewport.
9. **List group**: It is used to display an unordered series of content in a proper way.
10. **Card**: It provides a customizable, extensible, and flexible content container.
11. **Dropdown**: It is used to drop the menu in the format of a list of links, they are contextual and toggleable overlays.
12. **Navs**: It is used to create a basic and simple navigation menu with a .nav base class.
13. **Navbar**: The navigation bar is the headers at the top of a website or webpage.
14. **Forms**: Forms are used to take multiple inputs at once from the user. Bootstrap has two layouts available stacked and inline.

15. **Input groups**: They have extended form controls by adding a button, button group or text on either side of inputs.
16. **Breadcrumb**: It provides the location of the current page in a navigational hierarchy and also adds separators through CSS.
17. **Carousel**: It is a slide show of image or text content built with CSS 3D and JavaScript.
18. **Toast**: It displays a message for a small amount of time, a few seconds. They are alert messages designed to imitate push notifications popular in desktop and mobile systems.
19. **Tooltip**: It provides small information about the element/link when the mouse hovers over the element.
20. **Popovers**: It displays extra information about the element/link when clicked on it.
21. **Collapse**: It is a JavaScript plugin that is used to show or hide the content.
22. **Modal**: It is a small popup window positioned over the actual window.
23. **Pagination**: It is used to easily navigate between different pages, a large block of connected links is used for making them accessible.
24. **Media Object**: The Media object is used for repetitive and complex components like tweets or blogs. The images or videos are placed/aligned to the left or the right of the content.

