



**SRM Institute of Science and Technology**  
**College of Engineering and Technology**  
**School of Computing**

**DEPARTMENT OF COMPUTATIONAL INTELLIGENCE**

SRM Nagar, Kattankulathur – 603203, Chengalpattu District, Tamilnadu

**Academic Year: 2024-25 (EVEN)**

---

**18AIC302J**

**Web Programming for Artificial Intelligence**

**BOOTSTRAP**

# BOOTSTRAP

## Bootstrap Background and Features

Bootstrap is the most popular and powerful front-end (HTML, CSS, and JavaScript) framework for faster and easier responsive web development. It is also completely free, versatile, and intuitive.

Bootstrap is a front-end framework that helps you build mobile responsive websites more quickly and easily. First developed by Twitter, Bootstrap is by now used for anything from developing web applications to WordPress themes.

Bootstrap is a free and open-source tool collection for creating responsive websites and web applications. It is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first web sites. It solves many problems which we had once, one of which is the cross-browser compatibility issue. Nowadays, the websites are perfect for all the browsers (IE, Firefox and Chrome) and for all sizes of screens (Desktop, Tablets, Phablets, and Phones). All thanks to Bootstrap developers -Mark Otto and Jacob Thornton of Twitter, though it was later declared to be an open-source project.

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

## Need for Bootstrap

- Faster and Easier Web-Development.
- It creates Platform-independent web-pages.
- It creates Responsive Web-pages.
- It designed to be responsive to mobile devices too.
- It is Free! Available on [www.getbootstrap.com](http://www.getbootstrap.com)

## Advantages

- 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)
- Save lots of time — You can save lots of time and efforts using the Bootstrap predefined design templates and classes and concentrate on other development work.

- Consistent design — All Bootstrap components share the same design templates and styles through a central library, so the design and layout of your web pages will be consistent.
- Open Source — And the best part is, it is completely free to download and use.

## Getting Started with Bootstrap

There are two ways to include Bootstrap in the website.

- Include Bootstrap from CDN(Content Delivery Network) link.
- Download Bootstrap from [getbootstrap.com](https://getbootstrap.com) and use it.

How to use Bootstrap 4 in webpage:

1. It is a free front-end framework for faster and easier web development.
2. It combines HTML, CSS and JS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and numerous other components, as well as JavaScript plugins.
3. It provides you with a convenient approach for creating responsive designs.

With Bootstrap, you can conjure complex web pages from standard HTML and customize them to your needs. Bootstrap also comes with a number of jQuery plugins that can provide additional functionality such as carousels, buttons, tooltips, and more.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js"></script>
</head>
<body>
```

## Demystifying Grids Bootstrap Components

### Containers

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

There are two container classes to choose from:

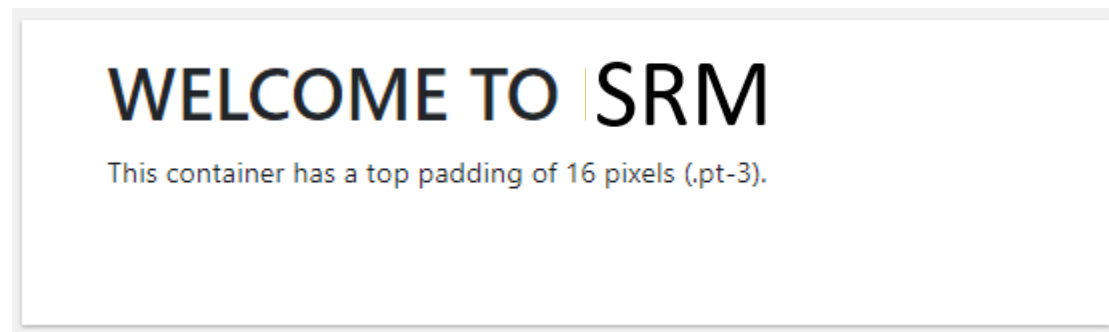
1. The **.container** class provides a responsive fixed width container
2. The **.container-fluid** class provides a full width container, spanning the entire width of the viewport.



### 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>
```



### Border and Color

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

```
<h1>HAVE A NICE DAY</h1>
```

```
<p>This container has a border and some extra padding and margins.</p>
```

</div>

# HAVE A NICE DAY

This container has a border and some extra padding and margins.

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

# HAVE A NICE DAY

This container has a dark background color and a white text, and some extra padding and margins.

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

# HAVE A NICE DAY

This container has a blue background color and a white text, and some extra padding and margins.

## Bootstrap 4 Grid System

Bootstrap's grid system is built with flexbox and allows up to 12 columns across the page.

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.

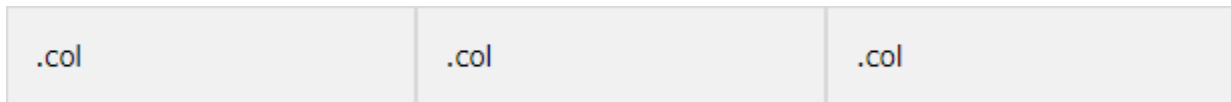
## Grid Classes

The Bootstrap 4 grid system has five classes:

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

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

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



## Basic Structure of a Bootstrap 4 Grid

The following is a 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>
```

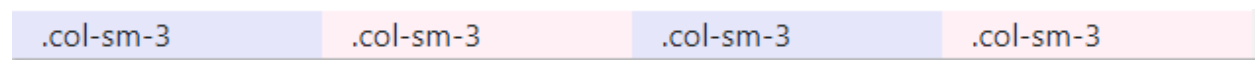
First example: create a row (`<div class="row">`). Then, add the desired number of columns (tags with appropriate `.col-*-*` classes). The first star (\*) represents the responsiveness:

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.

The following example shows how to create four equal-width columns starting at tablets and scaling to extra large desktops. On mobile phones or screens that are less than 576px wide, the columns will automatically stack on top of each other:

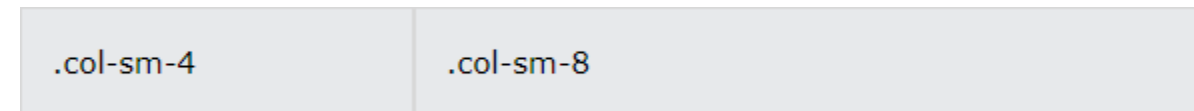
```
<div class="row">
  <div class="col-sm-3">.col-sm-3</div>
  <div class="col-sm-3">.col-sm-3</div>
  <div class="col-sm-3">.col-sm-3</div>
  <div class="col-sm-3">.col-sm-3</div>
</div>
```



### Two Unequal Responsive Columns

The following example shows how to get two various-width columns starting at tablets and scaling to large extra desktops:

```
<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:

**h1 Bootstrap heading** (2.5rem = 40px)

**h2 Bootstrap heading** (2rem = 32px)

**h3 Bootstrap heading** (1.75rem = 28px)

**h4 Bootstrap heading** (1.5rem = 24px)

**h5 Bootstrap heading** (1.25rem = 20px)

**h6 Bootstrap heading** (1rem = 16px)

#### 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

Display 1

Display 2

Display 3

Display 4

Activate Windows  
Go to PC settings

`<small>`

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

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

```
<h1>Lighter, Secondary Text</h1>
```



<p>The small element is used to create a lighter, secondary text in any heading:</p>  
<h1>h1 heading <small>secondary text</small></h1>  
<h2>h2 heading <small>secondary text</small></h2>  
<h3>h3 heading <small>secondary text</small></h3>  
<h4>h4 heading <small>secondary text</small></h4>  
<h5>h5 heading <small>secondary text</small></h5>  
<h6>h6 heading <small>secondary text</small></h6>  
</div>

**h1 heading** secondary text

**h2 heading** secondary text

**h3 heading** secondary text

**h4 heading** secondary text

**h5 heading** secondary text

**h6 heading** secondary text

<mark>

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

<p>Use the mark element to <mark>highlight</mark> text.</p>

Use the mark element to highlight text.

<abbr>

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

<p>The <abbr title="World Health Organization">WHO</abbr> was founded in 1948.</p>

The WHO was founded in 1948.

Text Colors

Bootstrap 4 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`:

This text is muted.

This text is important.

This text indicates success.

This text represents some information.

This text represents a warning.

This text represents danger.

Secondary text.

Dark grey text.

Body text.

Light grey text.

---

### 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`.

Note that background colors do not set the text color, so in some cases you'll want to use them together with a `.text-*` class.

This text is important.

This text indicates success.

This text represents some information.

This text represents a warning.

This text represents danger.

Secondary background color.

Dark grey background color.

Light grey background color.

### 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:

```
<div class="container">
  <h2>Basic Table</h2>
  <table class="table">
    <thead>
      <tr>
        <th>Firstname</th>
        <th>Lastname</th>
        <th>Email</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>AAAAA</td>
        <td>BBBBB</td>
        <td>AAAAABBB@example.com</td>
      </tr>
      <tr>
        <td>CCCCC</td>
        <td>DDDDDD</td>
        <td>CCCCCDDD@example.com</td>
      </tr>
      <tr>
        <td>EEEEEE</td>
      </tr>
    </tbody>
  </table>
</div>
```

```
<td>FFFFF</td>
<td>EEEEFF@example.com</td>
</tr>
</tbody>
</table>
</div>
```

## Basic Table

Firstname	Lastname	Email
AAAAA	BBBBB	AAAAABBB@example.com
CCCCC	DDDDDD	CCCCCDDD@example.com
EEEEE	FFFFF	EEEEFF@example.com

### Striped Rows

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

```
<table class="table table-striped">
```

Firstname	Lastname	Email
AAAAA	BBBBB	AAAAABBB@example.com
CCCCC	DDDDDD	CCCCCDDD@example.com
EEEEE	FFFFF	EEEEFF@example.com

### 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:

### 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>).

## Bootstrap 4 Images

### Rounded Corners

The .rounded class adds rounded corners to an image:

```

```

### Circle

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

```

```

### Thumbnail

The .img-thumbnail class shapes the image to a thumbnail (bordered):

```

```

Rounded Corners:



Circle:



Thumbnail:



## Aligning Images

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

```

```

## Centered Image

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

```

```

## Bootstrap 4 Jumbotron

A jumbotron indicates a big grey box for calling extra attention to some special content or information.

```
<div class="container">  
  <div class="jumbotron">  
    <h1>Bootstrap Tutorial</h1>
```

`<p>Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile-first projects on the web.</p>`

`</div>`

`<p>This is some text.</p>`

`<p>This is another text.</p>`

`</div>`

# Bootstrap Tutorial

Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile-first projects on the web.

This is some text.

This is another text.

## Full-width Jumbotron

If you want a full-width jumbotron without rounded borders, add the `.jumbotron-fluid` class and a `.container` or `.container-fluid` inside of it:

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

## Bootstrap 4 Alerts

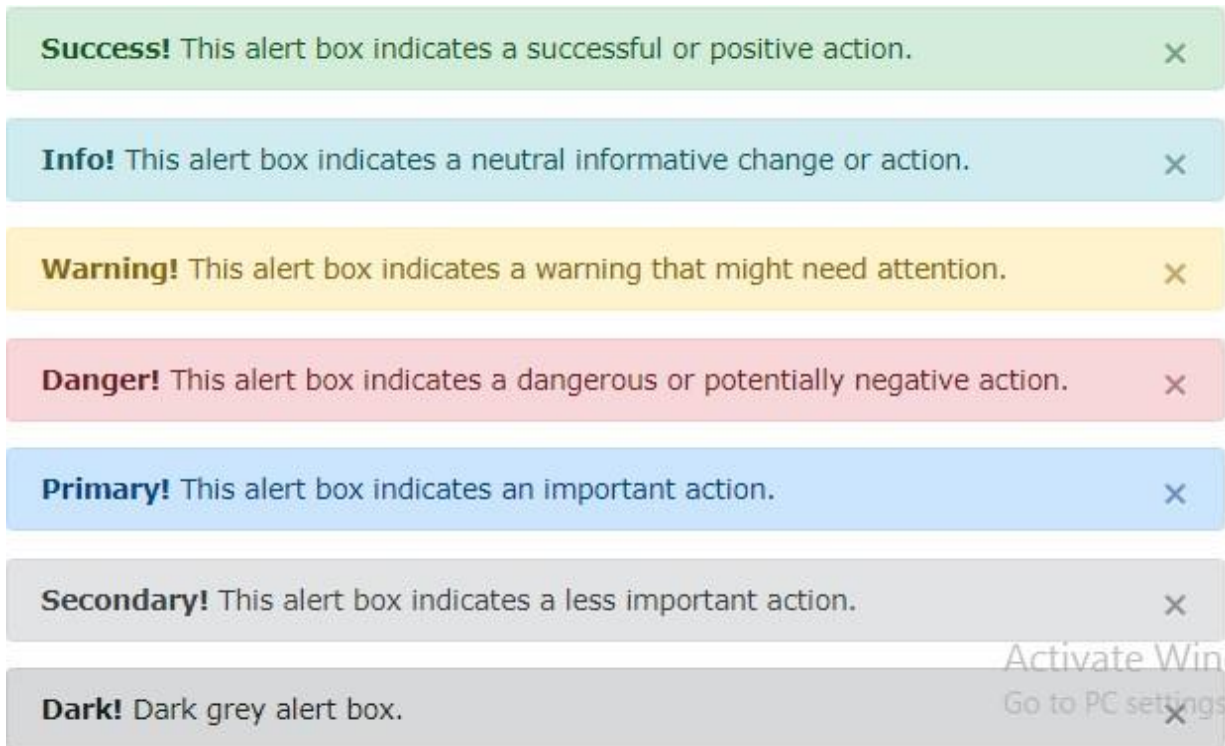
Bootstrap 4 provides an easy way to create predefined alert messages:

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>
```



### Alert Links

Add the alert-link class to any links inside the alert box to create "matching colored links":

```
<div class="alert alert-success">
  <strong>Success!</strong> You should <a href="#" class="alert-link">read this message</a>.
</div>
```

### Bootstrap 4 Buttons

Bootstrap 4 provides different styles of buttons:

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



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

### Example

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

### Button Outline

Bootstrap 4 provides eight outline/bordered buttons:

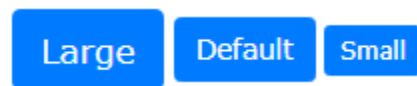
```
<button type="button" class="btn btn-outline-primary">Primary</button>
```



### 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>
<button type="button" class="btn btn-primary">Default</button>
<button type="button" class="btn btn-primary btn-sm">Small</button>
```



### Active/Disabled Buttons

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

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

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

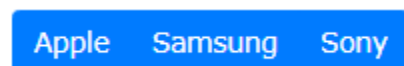
### Spinner Buttons

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

### Button Groups

Bootstrap 4 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:

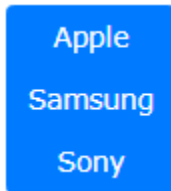
### Vertical Button Groups



Bootstrap 4 also supports vertical button groups:

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

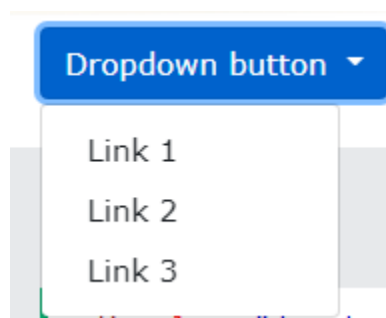
```
<div class="btn-group-vertical">  
  <button type="button" class="btn btn-primary">Apple</button>  
  <button type="button" class="btn btn-primary">Samsung</button>  
  <button type="button" class="btn btn-primary">Sony</button>  
</div>
```



### 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-toggle="dropdown">  
    Dropdown button  
  </button>  
  <div class="dropdown-menu">  
    <a class="dropdown-item" href="#">Link 1</a>  
    <a class="dropdown-item" href="#">Link 2</a>  
    <a class="dropdown-item" href="#">Link 3</a>  
  </div>  
</div>
```



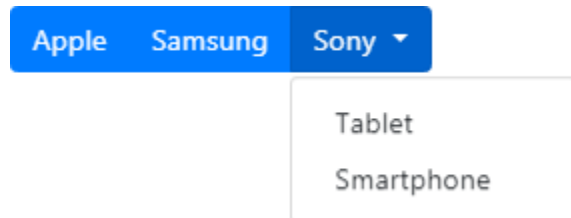
### Nesting Button Groups & Dropdown Menus

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

```

</button>
<div class="dropdown-menu">
  <a class="dropdown-item" href="#">Tablet</a>
  <a class="dropdown-item" href="#">Smartphone</a>
</div>
</div>
</div>

```



### Bootstrap 4 Progress Bars

A progress bar can be used to show a user how far along he/she 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:50%"></div>
</div>

```



The height of the progress bar is 16px by default. Use the CSS height property to change it.

```

<div class="progress" style="height:30px">
  <div class="progress-bar" style="width:60%;height:30px">60%</div>
</div>
</div>

```



### Colored Progress Bars

By default, the progress bar is blue (primary). Use any of the Bootstrap 4 contextual background classes to its color:

```

<!-- Green -->
<div class="progress">
  <div class="progress-bar bg-success" style="width:20%"></div>
</div>

```

### Striped Progress Bars

Use the `.progress-bar-striped` class to add stripes to the progress bars:

```

<div class="progress">
  <div class="progress-bar bg-warning progress-bar-striped" style="width:60%"></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

```
<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>
```



## Menus and Navigations

### Bootstrap 4 Navs

#### Nav Menus

to create a simple horizontal menu, add the .nav class to a <ul> element, followed by .nav-item for each <li> 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>
  <li class="nav-item">
    <a class="nav-link" href="#">Link</a>
  </li>
  <li class="nav-item">
    <a class="nav-link disabled" href="#">Disabled</a>
  </li>
</ul>
```

[Link](#) [Link](#) [Link](#) [Disabled](#)

### Aligned Nav

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

```
<!-- Centered nav -->
```

```
<ul class="nav justify-content-center">
```

```
<!-- Right-aligned nav -->
```

```
<ul class="nav justify-content-end">
```

### Vertical Nav

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

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

[Link](#)

[Link](#)

[Link](#)

[Disabled](#)

### 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" href="#">Link</a>
```

```
</li>
```

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

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

```
</li>
```

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

```
<a class="nav-link disabled" href="#">Disabled</a>
```

```
</li>
```

```
</ul>
```

[Active](#)

[Link](#)

[Link](#)

[Disabled](#)

### Pills

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

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



### Justified Tabs/pills

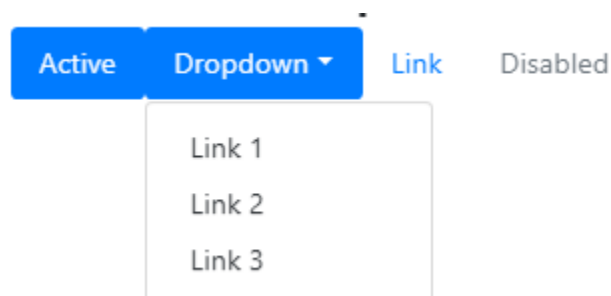
Justify the tabs/pills with the `.nav-justified` class (equal width):

```
<ul class="nav nav-pills nav-justified">..</ul>
```

```
<ul class="nav nav-tabs nav-justified">..</ul>
```

### Pills with Dropdown

```
<ul class="nav nav-pills">
  <li class="nav-item">
    <a class="nav-link active" href="#">Active</a>
  </li>
  <li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle" data-toggle="dropdown" href="#">Dropdown</a>
    <div class="dropdown-menu">
      <a class="dropdown-item" href="#">Link 1</a>
      <a class="dropdown-item" href="#">Link 2</a>
      <a class="dropdown-item" href="#">Link 3</a>
    </div>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Link</a>
  </li>
  <li class="nav-item">
    <a class="nav-link disabled" href="#">Disabled</a>
  </li>
</ul>
```



### Toggleable / Dynamic Tabs

To make the tabs toggleable, add the data-toggle="tab" attribute to each link. Then add a .tab-pane class with a unique ID for every tab and wrap them inside a <div> element with class .tab-content.

If you want the tabs to fade in and out when clicking on them, add the .fade class to .tab-pane:

```
<a class="nav-link" data-toggle="tab" href="#menu1">Vision</a>
</li>
<li class="nav-item">
  <a class="nav-link" data-toggle="tab" href="#menu2">Mission</a>
</li>
</ul>

<!-- Tab panes -->
<div class="tab-content">
  <div id="home" class="container tab-pane active"><br>
    <h3>HOME</h3>
    <p>Rajalakshmi Engineering College, an autonomous institution affiliated to Anna University, Chennai, was established in the year 1997 under the aegis of Rajalakshmi Educational Trust whose members have had consummate experience in the fields of education and industry.</p>
  </div>
  <div id="menu1" class="container tab-pane fade"><br>
    <h3>Vision</h3>
    <p>To be an institution of excellence in Engineering, Technology and Management Education & Research. To provide competent and ethical professionals with a concern for society.</p>
  </div>
  <div id="menu2" class="container tab-pane fade"><br>
    <h3>Mission</h3>
    <p>To impart quality technical education imbued with proficiency and humane values. To provide right ambience and opportunities for the students to develop into creative, talented and globally competent professionals.</p>
  </div>
</div>
</div>
```

# Toggleable Tabs

Home

Vision

Mission

## HOME

Rajalakshmi Engineering College, an autonomous institution affiliated to Anna University, Chennai, was established in the year 1997 under the aegis of Rajalakshmi Educational Trust whose members have had consummate experience in the fields of education and industry.

### Navigation Bars

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

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-xl|lg|md|sm` (stacks the navbar vertically on extra large, large, medium or small screens).

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

### Bootstrap 4 Forms

#### Bootstrap 4's Default Settings

Form controls automatically receive some global styling with Bootstrap:

All textual `<input>`, `<textarea>`, and `<select>` elements with class `.form-control` have a width of 100%.

#### Bootstrap 4 Form Layouts

Bootstrap provides two types of form layouts:

1. Stacked (full-width) form
2. Inline form

Add a wrapper element with `.form-group`, around each form control, to ensure proper margins:

#### Bootstrap 4 Stacked Form

The following example creates a stacked form with two input fields, one checkbox, and a submit button.

Add a wrapper element with `.form-group`, around each form control, to ensure proper margins:

```
<form action="/action_page.php">
  <div class="form-group">
    <label for="email">Email address:</label>
```

```

    <input type="email" class="form-control" placeholder="Enter email" id="email">
</div>
<div class="form-group">
    <label for="pwd">Password:</label>
    <input type="password" class="form-control" placeholder="Enter password" id="pwd">
</div>
<div class="form-group form-check">
    <label class="form-check-label">
        <input class="form-check-input" type="checkbox"> Remember me
    </label>
</div>
<button type="submit" class="btn btn-primary">Submit</button>
</form>

```

Email:

Password:

☐ Remember me

### Bootstrap Inline Form

In an inline form, all of the elements are inline and left-aligned.

Note: This only applies to forms within viewports that are at least 576px wide. On screens smaller than 576px, it will stack horizontally.

Additional rule for an inline form:

Add class `.form-inline` to the `<form>` element

The following example creates an inline form with two input fields, one checkbox, and one submit button:

Example

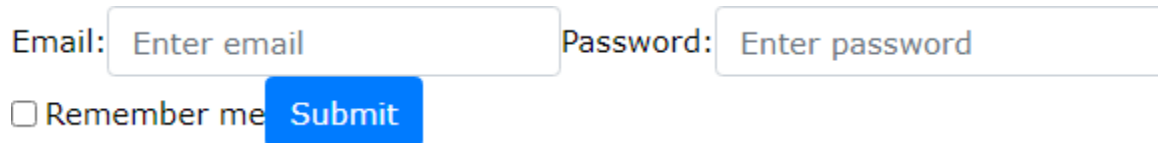
```

<form class="form-inline" action="/action_page.php">
    <label for="email">Email address:</label>
    <input type="email" class="form-control" placeholder="Enter email" id="email">
    <label for="pwd">Password:</label>
    <input type="password" class="form-control" placeholder="Enter password" id="pwd">

```



```
<div class="form-check">
  <label class="form-check-label">
    <input class="form-check-input" type="checkbox"> Remember me
  </label>
</div>
<button type="submit" class="btn btn-primary">Submit</button>
</form>
```



In an inline form, all of the elements are inline and left-aligned.

Note: This only applies to forms within viewports that are at least 576px wide. On screens smaller than 576px, it will stack horizontally.

Additional rule for an inline form:

Add class .form-inline to the <form> element

## Supported Form Controls

Bootstrap supports the following form controls:

- input
- textarea
- checkbox
- radio
- select

## Bootstrap Input

Bootstrap supports all the HTML5 input types: text, password, datetime, datetime-local, date, month, time, week, number, email, url, search, tel, and color.

The following example contains two input elements; one of type="text" and one of type="password". As we mentioned in the Forms chapter, we use the .form-control class to style inputs with full-width and proper padding, etc:

```
<div class="form-group">
  <label for="usr">Name:</label>
  <input type="text" class="form-control" id="usr">
</div>
<div class="form-group">
  <label for="pwd">Password:</label>
  <input type="password" class="form-control" id="pwd">
</div>
```

### Bootstrap Textarea

```
<div class="form-group">
  <label for="comment">Comment:</label>
  <textarea class="form-control" rows="5" id="comment"></textarea>
</div>
```

### Bootstrap Checkboxes

Use a wrapper element with class="form-check" to ensure proper margins for labels and checkboxes.

Add the .form-check-label class to label elements, and .form-check-input to style checkboxes properly inside the .form-check container.

```
<div class="form-check">
  <label class="form-check-label">
    <input type="checkbox" class="form-check-input" value="">Option 1
  </label>
</div>
<div class="form-check">
  <label class="form-check-label">
    <input type="checkbox" class="form-check-input" value="">Option 2
  </label>
</div>
<div class="form-check">
  <label class="form-check-label">
    <input type="checkbox" class="form-check-input" value="" disabled>Option 3
  </label>
</div>
```

### Inline Checkboxes

```
<div class="form-check-inline">
```

### Bootstrap Radio Buttons

```
<div class="form-check">
  <label class="form-check-label">
    <input type="radio" class="form-check-input" name="optradio">Option 1
  </label>
</div>
<div class="form-check">
  <label class="form-check-label">
    <input type="radio" class="form-check-input" name="optradio">Option 2
  </label>
</div>
<div class="form-check disabled">
  <label class="form-check-label">
```

```
<input type="radio" class="form-check-input" name="optradio" disabled>Option 3
</label>
</div>
```

### Bootstrap Select List

```
<div class="form-group">
  <label for="sel1">Select list:</label>
  <select class="form-control" id="sel1">
    <option>1</option>
    <option>2</option>
    <option>3</option>
    <option>4</option>
  </select>
</div>
```

### Form Control File and Range

Add the `.form-control-range` class to input type "range" or `.form-control-file` to input type "file" to style a range control or a file field with full-width:

```
<input type="range" class="form-control-range">
<input type="file" class="form-control-file border">
```

### Form Validation

different validation classes to provide valuable feedback to users. Add either `.was-validated` or `.needs-validation` to the `<form>` element, depending on whether you want to provide validation feedback before or after submitting the form. The input fields will have a green (valid) or red (invalid) border to indicate what's missing in the form. You can also add a `.valid-feedback` or `.invalid-feedback` message to tell the user explicitly what's missing, or needs to be done before submitting the form.

```
<form action="/action_page.php" class="was-validated">
  <div class="form-group">
    <label for="uname">Username:</label>
    <input type="text" class="form-control" id="uname" placeholder="Enter username"
name="uname" required>
    <div class="valid-feedback">Valid.</div>
    <div class="invalid-feedback">Please fill out this field.</div>
  </div>
  <div class="form-group">
    <label for="pwd">Password:</label>
    <input type="password" class="form-control" id="pwd" placeholder="Enter password"
name="pswd" required>
    <div class="valid-feedback">Valid.</div>
    <div class="invalid-feedback">Please fill out this field.</div>
  </div>
  <div class="form-group form-check">
    <label class="form-check-label">
```

```
<input class="form-check-input" type="checkbox" name="remember" required> I agree.  
<div class="valid-feedback">Valid.</div>  
<div class="invalid-feedback">Check this checkbox to continue.</div>  
</label>  
</div>  
<button type="submit" class="btn btn-primary">Submit</button>  
</form>
```

## Plugins

### Extending Using Bootstrap Plugins

Bootstrap comes with a huge bundle of reusable components and jQuery plugins that we can use into our projects. The great thing about Bootstrap is that we can choose to use them or not. And even better, we can extend its functionality using plugins and addons. And there are hundreds of them. You just need to find out which suits you the best.

Here are a list of some of the plugins and addons.

### Galleries

- Bootstrap Lightbox
- Simple Lightbox
- Bootstrap Image Gallery
- Codetabs
- Bootslider
- Strapslide

### Dialogs and Notifications

- Bootbox.js
- Bootstrap Modal

- Bootstrap Grownl
- BootStrap Notify

#### Form Elements

- Bootstrap Form Helpers
- Bootstrap Tags
- Bootstrap Switch

#### Date and Time Pickers

- Daterange Picker
- Bootstrap TimePicker
- ClockFace
- Bootstrap DateTime
- Bic-Calendar
- ClockPicker
- Calendar

#### Color Pickers

- Pick a Color
- Color Picker for Bootstrap
- Color Palatte

#### Tables

- TableCloth
- Data Tables

#### Buttons and icons

- Icon Animator
- CSS3 Microsoft Modern Buttons
- Bootstrap Buttons Bundle
- Social Buttons

#### Navigation and Menus

- Yamm!
- Bootstrap Sidebar
- Bootstrap TreeNav
- 3D Mega Menus

## Flexbox & Layouts.

### Flexbox

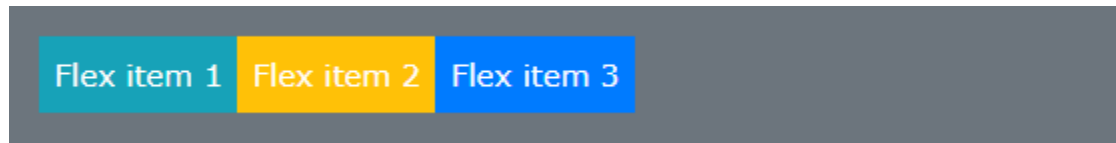
The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning.

**Note: Flexbox is not supported in IE9 and earlier versions.**

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

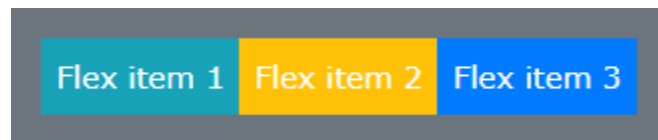
```
<div class="d-flex p-3 bg-secondary text-white">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
  <div class="p-2 bg-primary">Flex item 3</div>
```

</div>



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

```
<div class="d-inline-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>
```



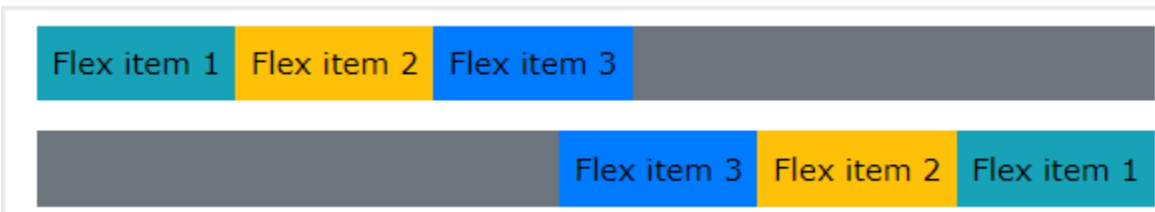
### Horizontal Direction

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:

```
<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:

```
<div class="d-flex flex-column">  
  <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-column-reverse">  
  <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>
```

Flex item 1

Flex item 2

Flex item 3

Flex item 3

Flex item 2

Flex item 1

### Justify Content

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

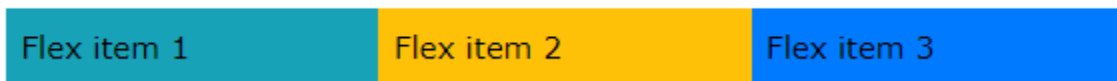
```
<div class="d-flex justify-content-start">...</div>  
<div class="d-flex justify-content-end">...</div>  
<div class="d-flex justify-content-center">...</div>  
<div class="d-flex justify-content-between">...</div>  
<div class="d-flex justify-content-around">...</div>
```



### Fill / Equal Widths

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

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

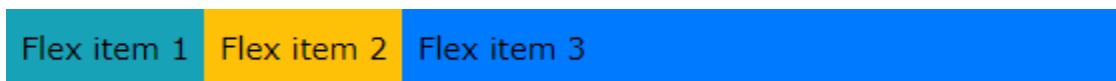


### Grow

Use `.flex-grow-1` on a flex item to take up the rest of the space. In the example below, the first two flex items take up their necessary space, while the last item takes up the rest of the available space:

```
<div class="d-flex">  
  <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-grow-1">Flex item 3</div>  
</div>
```

Tip: Use `.flex-shrink-1` on a flex item to make it shrink if necessary.

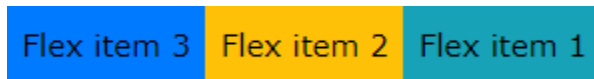


### Order



Change the visual order of a specific flex item(s) with the `.order` classes. Valid classes are from 0 to 12, where the lowest number has highest priority (order-1 is shown before order-2, etc..):

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



### Auto Margins

Easily add auto margins to flex items with `.mr-auto` (push items to the right), or by using `.ml-auto` (push items to the left):

```
<div class="d-flex bg-secondary">
  <div class="p-2 mr-auto 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 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 ml-auto bg-primary">Flex item 3</div>
</div>
```



### 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`.

### 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).

### Fixed Layout with Bootstrap

With Bootstrap 4 you can still create web page layouts based on fixed number of pixels, however the container width vary depending on the viewport width and the layout is responsive too.

The process of creating the fixed yet responsive layout basically starts with the `.container` class. After that you can create rows with the `.row` class to wrap the horizontal groups of columns. Rows must be placed within a `.container` for proper alignment and padding.

Further columns can be created inside a row using predefined grid classes like `.col-*`, `.col-sm-*`, `.col-md-*`, `.col-lg-*` and `.col-xl-*` where `*` represent grid number and should be from 1 to 12. Please check out the Bootstrap grid system tutorial to learn more about grid classes.

#### Fluid Layout

In Bootstrap you can use the class `.container-fluid` to create fluid layouts to utilize the 100% width of the viewport across all device sizes (extra small, small, medium, large, and extra large).

The class `.container-fluid` simply applies the width: 100% instead of different width for different viewport sizes. However, the layout will still responsive and you can use the grid classes as usual. See the tutorial on Bootstrap grid system to learn more about grid classes.