

# Web Programming Technologies

Harshita Maheshwari

# Session-6

# Topics to be covered.....

- What is Responsive Web Design?
- Responsive web design-Frameworks
- Overview and Need to use Bootstrap
- Bootstrap Grid System
- Bootstrap Typography
- Bootstrap tables
- Images
- Alerts
- Badge
- List Groups
- Buttons and Button Groups
- Bootstrap Cards
- Dropdowns
- Collapse
- Modal

# Responsive Web Designing (RWD)

Responsive Web Designing (RWD) is a process of designing a single website to be used and compatible on different portable or handy electronic devices. Also known as **Adaptive Web Designing (AWD)**

It regarded as an integrated approach of designing through which compelling and easy to use websites are built, to give an optimal viewing user experience across a wide variety of devices starting from desktop computers to mobile phones

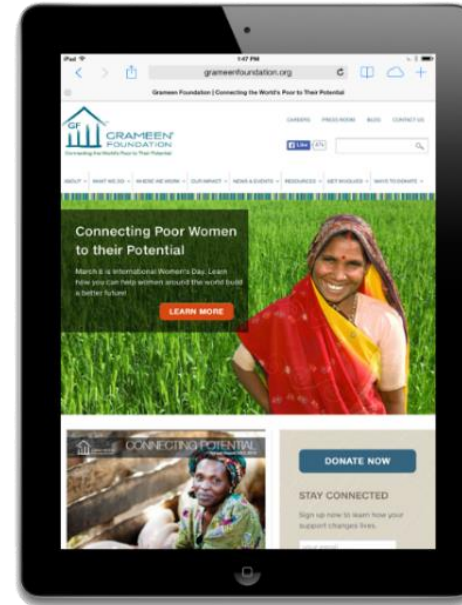
## Responsive Websites Offers:

- ✓ Smooth navigation
- ✓ Easy reading
- ✓ Minimum pinching
- ✓ Reduces scrolling and zooming.
- ✓ Excellent user experience.

# State of today's Web





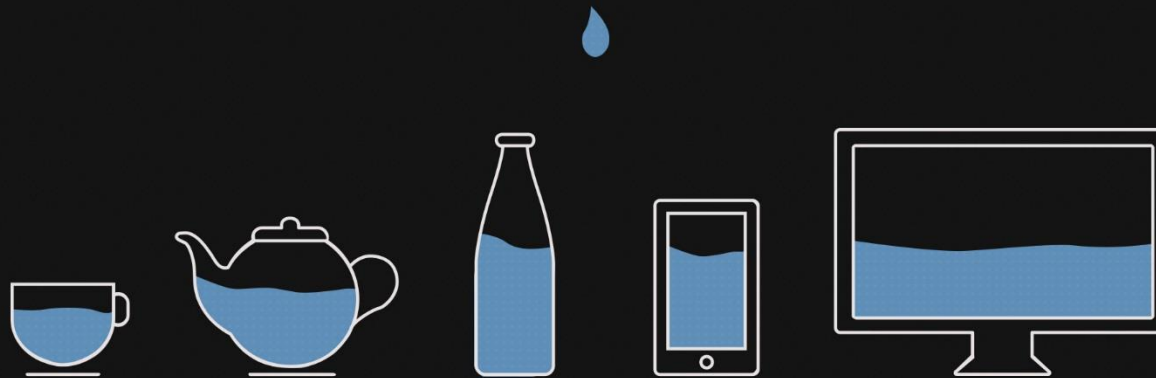


# Need of Responsive Web Design

- Growing Demand for Smartphones
- Multiple Screen Sizes and Mobile Browsers
- Wide Usage of Internet
- Permits wider browser support
- Compulsory for Getting Good Business



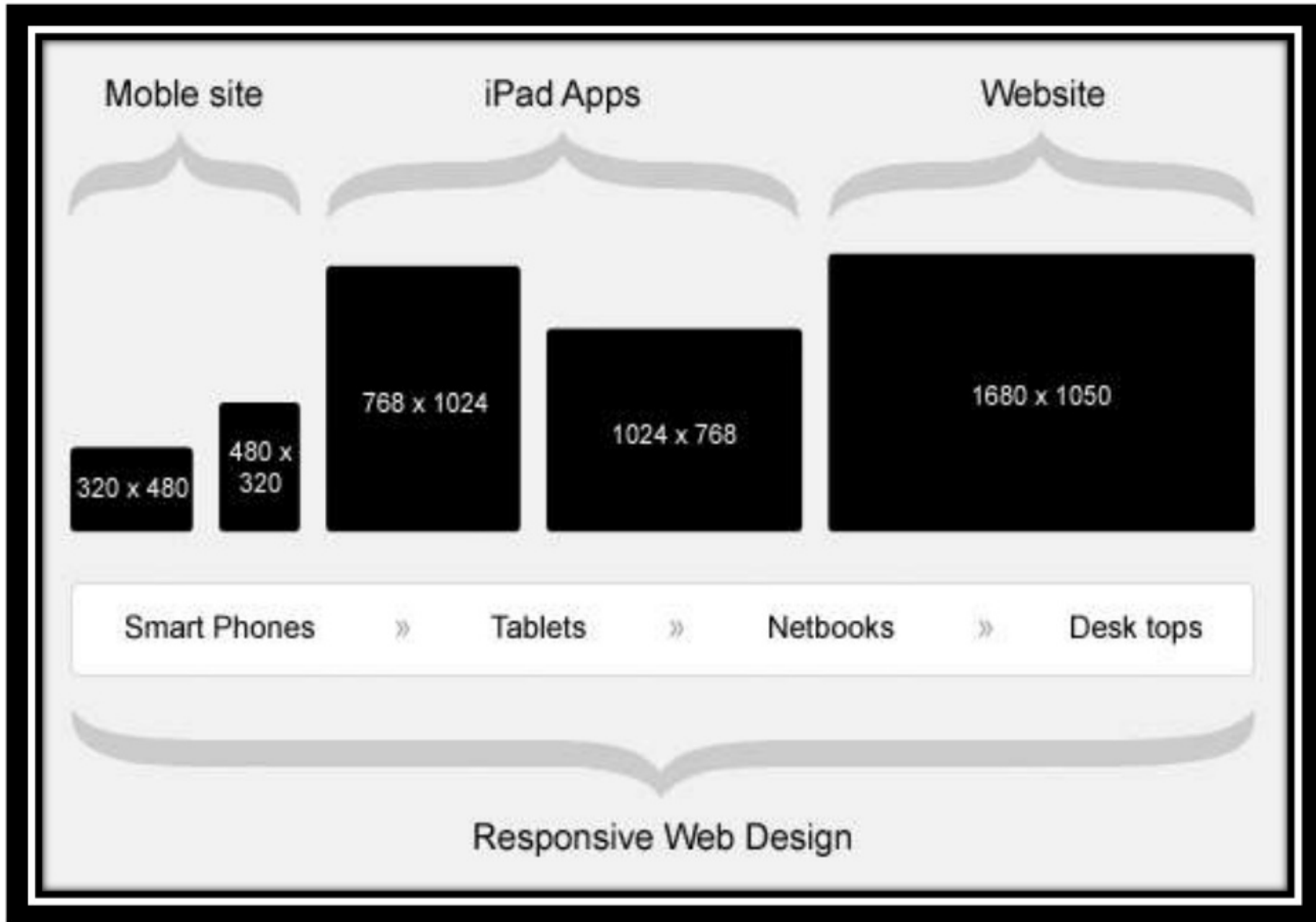
## CONTENT IS LIKE WATER



“ You put water into a cup it becomes the cup.  
 You put water into a bottle it becomes the bottle.  
 You put it in a teapot, it becomes the teapot. ”

Josh Clark (originally Bruce Lee) - Seven deadly mobile myths

Illustration by Stéphanie Walter



# What is Responsive Web Design ?



One URL



Single content



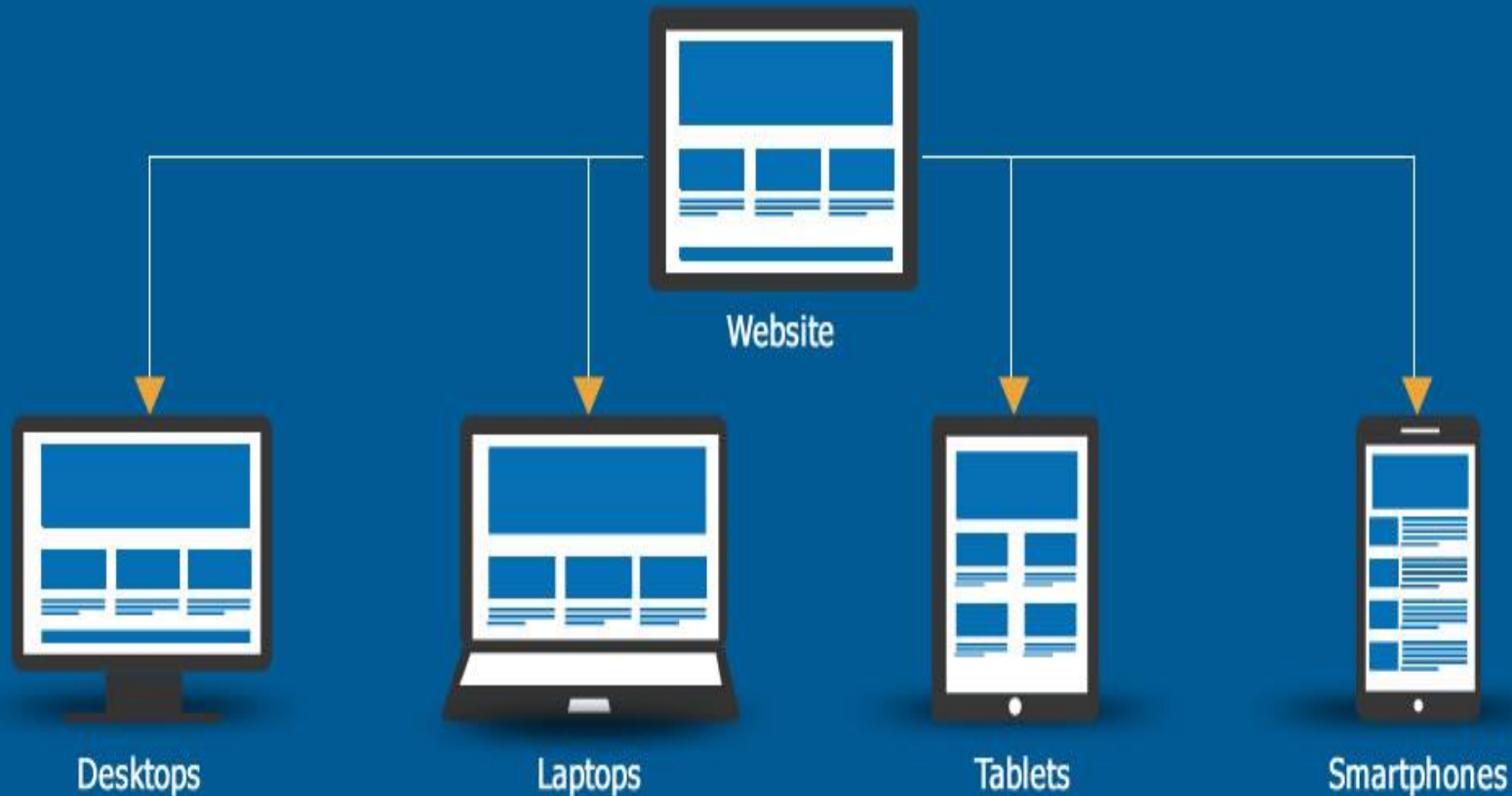
Same Code



Using CSS3  
Media Queries



Multi screen



A freely available design framework for websites  
and web applications



Based upon HTML5, CSS and JavaScript

Released on GitHub in August 2011

Developed by Mark Otto and Jacob Thornton at  
Twitter.

Bootstrap is the most popular framework for RAPID  
WEBSITE PROTOTYPING..

# Why use Bootstrap

## ➤ Mobile first approach:

Since Bootstrap 3, the framework consists of Mobile first styles throughout the entire library instead of in separate files.

## ➤ Browser Support:

It is supported by all popular browsers.

## ➤ Easy to get started:

With just the knowledge of HTML and CSS anyone can get started with Bootstrap. Also the Bootstrap official site has a good documentation.

## ➤ Responsive design:

Bootstrap's responsive CSS adjusts to Desktops, Tablets and Mobiles.

# What's new in Bootstrap 5 (in compare to BS4)

- **jQuery isn't required anymore** (however you can still use jQuery methods if you wish)
- Dropped support for **Internet Explorer**
- The use of JavaScript is **minimized in favor of CSS (CSS custom properties)**
- Almost each option is **available as data-attribute** (it can be set manually without JS)
- Enhanced **Grid system**
- Enhanced **modularity**
- Enhanced **customization**
- New components/helpers/utilities/variations
- Some components were removed (i.e. Jumbotron)
- Enhanced **Icons** (SVG Icon Library)

# Where to Get Bootstrap?

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

- **BootstrapCDN** is a free and public *Content Delivery Network*. Users of BootstrapCDN can load CSS, JavaScript and images remotely, from its servers.

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta2/dist/css/bootstrap.min.css"
rel="stylesheet">
```

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta2/dist/js/bootstrap.min.js"></script>
```

OR

- We can download the latest version of Bootstrap from <http://getbootstrap.com/>.



Once the compiled version Bootstrap is downloaded, extract the ZIP file, and you will see the following file/directory structure:

```
bootstrap/
├── css/
│   ├── bootstrap-grid.css
│   ├── bootstrap-grid.css.map
│   ├── bootstrap-grid.min.css
│   ├── bootstrap-grid.min.css.map
│   ├── bootstrap-reboot.css
│   ├── bootstrap-reboot.css.map
│   ├── bootstrap-reboot.min.css
│   ├── bootstrap-reboot.min.css.map
│   ├── bootstrap.css
│   ├── bootstrap.css.map
│   ├── bootstrap.min.css
│   └── bootstrap.min.css.map
└── js/
    ├── bootstrap.bundle.js
    ├── bootstrap.bundle.js.map
    ├── bootstrap.bundle.min.js
    ├── bootstrap.bundle.min.js.map
    ├── bootstrap.js
    ├── bootstrap.js.map
    ├── bootstrap.min.js
    └── bootstrap.min.js.map
```



Bootstrap grid system is used for creating page layout through a series of rows and columns.

The grid system consists of 12 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											

## Bootstrap Grid Classes:

Device	Class
Extra small (<576px)	.col-
Small (≥576px)	.col-sm-
Medium (≥768px)	.col-md-
Large (≥992px)	.col-lg-
Extra large (≥1200px)	.col-xl-
Xxl (≥1400px)	.col-xxl-

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

# Bootstrap Grid System Example: Medium and Extra Small Device

Here we had used 2 div's and gave them the 50%/50% split at the medium and extra small viewport width:

```
<div class="container">
  <div class="row">
    <div class="col-md-6 col-4">
      <p>div 1...</p>
      <p>lgrid1 ...</p>
    </div>
    <div class="col-md-6 col-8">
      <p>div2...</p>
      <p>grid2...</p>
    </div>
  </div>
</div>
```

Bootstrap requires a containing element to wrap site contents and house our grid system.

Use `.container` for a responsive fixed width container.

Use `.container-fluid` for a full width container, spanning the entire width of your viewport.

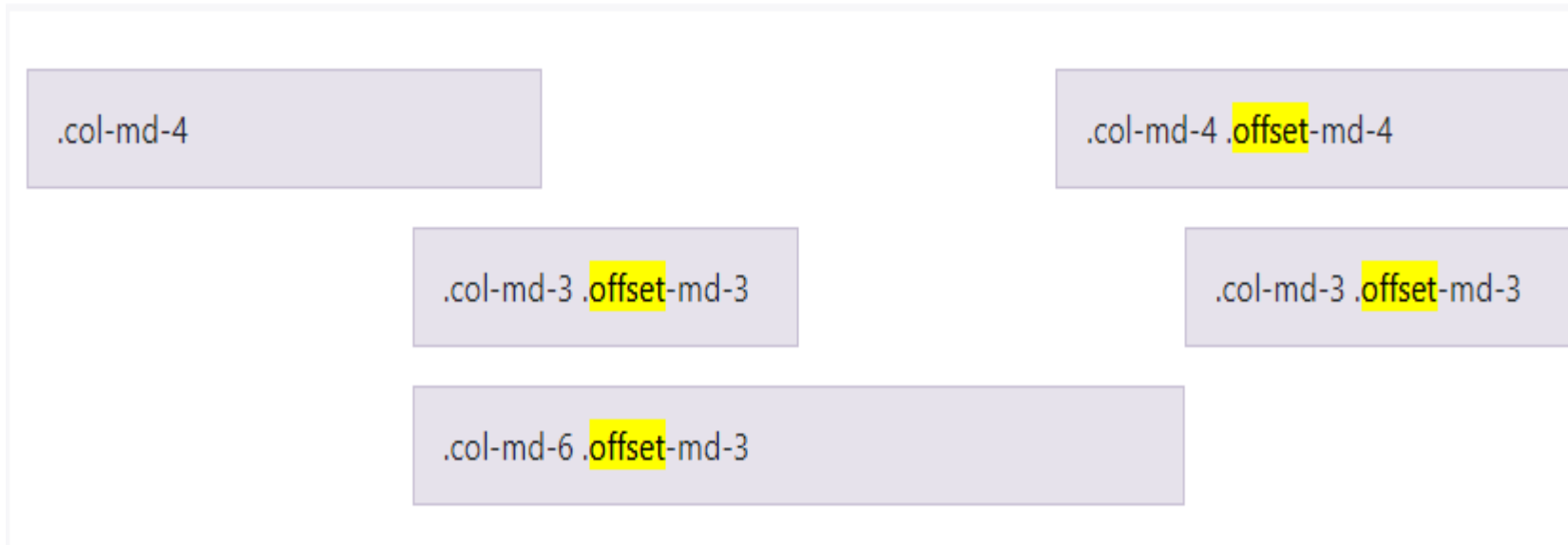
```
div class="container">
    ...
</div>

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

# Bootstrap Grid column Offset

Move columns to the right.

These classes increase the left margin of a column by \* columns.



Make it easy to create heading, paragraph, ol, ul, inline element etc. in a way that would be appealing to the users.

```
<span class="h1">H1</span>
<span class="h2">H2</span>
<span class="h3">H3</span>
<span class="h4">H4</span>
<span class="h5">H5</span>
<span class="h6">H6</span>
```

Output: **H1** H2 H3 H4 H5 H6

```
<h1>Main Heading 1 <span class="small">Small Heading 1</span></h1>
```

OR

```
<h1>Main Heading 1 <small>Small Heading 1</small></h1>
```

Output:

**Main Heading 1** Small Heading 1

**Display Headings:** used to stand out more than normal headings (larger font-size and lighter font-weight)

```
<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>
```

## More Typography Classes:

.text-start

Indicates left-aligned text

.text-center

Indicates center-aligned text

.text-end

Indicates right-aligned text

.text-justify

Indicates justified text

.text-nowrap

Indicates no wrap text

.text-lowercase

Indicates lowercased text

.text-uppercase

Indicates uppercased text

.text-capitalize

Indicates capitalized text

.font-weight-light

Light weight text

.font-weight-bold

Bold text.

font-italic

Italic text

## Text Colors:

.text-muted, .text-primary,  
.text-success, .text-info,  
.text-warning, .text-danger,  
.text-secondary, .text-dark,  
.text-light, .text-body,  
.text-light, .text-white

## Background Colors:

.bg-primary, .bg-success,  
.bg-info, .bg-warning,  
.bg-danger, .bg-secondary,  
.bg-dark, .bg-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.

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 classes for styling tables:

Class	Purpose
.table	adds basic styling to a table
.table-striped	adds zebra-stripes to a table
.table-bordered	adds borders on all sides of the table and cells
.table-hover	adds a hover effect (grey background color) on table rows
.table-dark	adds a black background to the table
.table-borderless	removes borders from the table

## Contextual classes to color table rows or individual cells:

Class	Description
<code>.table-primary</code>	Blue: Indicates an important action
<code>.table-success</code>	Green: Indicates a successful or positive action
<code>.table-danger</code>	Red: Indicates a dangerous or potentially negative action
<code>.table-info</code>	Light blue: Indicates a neutral informative change or action
<code>.table-warning</code>	Orange: Indicates a warning that might need attention
<code>.table-active</code>	Grey: Applies the hover color to the table row or table cell
<code>.table-secondary</code>	Grey: Indicates a slightly less important action
<code>.table-light</code>	Light grey table or table row background
<code>.table-dark</code>	Dark grey table or table row background

```
<div class="table-responsive">
  <table class="table table-bordered table-hover">
    <!--Html code-->
  </table>
</div>
```

# Bootstrap Images

Rounded Corners:



Circle:



Thumbnail:



Class	Purpose
.rounded	adds rounded corners to an image
.rounded-circle	shapes the image to a circle
.img-thumbnail	shapes the image to a thumbnail (bordered)
.float-end	Float an image to the right
.float-start	Float an image to the left
.img-fluid	Create responsive image.(applies max-width: 100%; and height: auto; to the image)
.mx-auto .d-block	Center the image

# Bootstrap alert

Creating an alert with Bootstrap is easy. All we need to do is create a `<div>` element with classes `alert` and one of the following contextual state classes.

`.alert-success`, `.alert-info`, `.alert-warning`,  
`.alert-danger`, `.alert-primary`, `.alert-secondary`, `.alert-light` or `.alert-dark`

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

`.alert-dismissible` - To close the alert message, add a class to the alert container. Then add `class="close"` and `data-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> Indicates a successful or positive action.
</div>
```



# Badges

Badges are used to add additional information to any content.

Ex:

```
<h1>HTML<span class="badge bg-secondary">New</span></h1>
```

## Contextual Badges:



## Badge inside an Element

```
<button type="button" class="btn btn-primary">
  Messages <span class="badge bg-light">4</span>
</button>
```



## Button Styles:

`.btn, .btn-primary, .btn-secondary, .btn-success, .btn-info, .btn-warning, .btn-danger, .btn-dark, .btn-light, .btn-link`



Button classes can be used on
<code>&lt;a&gt;</code>
<code>&lt;button&gt;</code>
<code>&lt;input type="button"&gt;</code>
<code>&lt;input type="submit"&gt;</code>

```
<button class="btn">Button</button>
```

```
<p><button class="btn btn-primary">
btn-primary</button></p>
```

**To make an anchor element look like a button,** use `btn` class along with the other button classes (like `btn-default`, `btn-primary` etc.)

```
<a href="#" class="btn btn-info" role="button">Link Button</a>
```

## Button Outline:

`.btn-outline-primary, . btn-outline-secondary, . btn-outline-success, . btn-outline-info, . btn-outline-warning, . btn-outline-danger, . btn-outline-dark, . btn-outline-light`



## Block Level Buttons:

```
<div class="d-grid gap-2">
<button class="btn btn-primary" type="button">Button</button>
</div>
```

Active/Disabled Buttons: `.active, .disabled`

Button Sizes: `.btn-lg, .btn-sm`

# Bootstrap Button Group

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

```
<div class="btn-group">
  <button type="button" class="btn btn-primary">Button1</button>
  <button type="button" class="btn btn-primary">Button2</button>
  <button type="button" class="btn btn-primary">Button3</button>
</div>
```

.btn-group-lg|sm can also be used to size all the buttons in the group.

.btn-group-vertical to create a vertical button group.



A card in Bootstrap 4 is a bordered box with some padding around its content. It includes options for headers, footers, content, colors, etc.



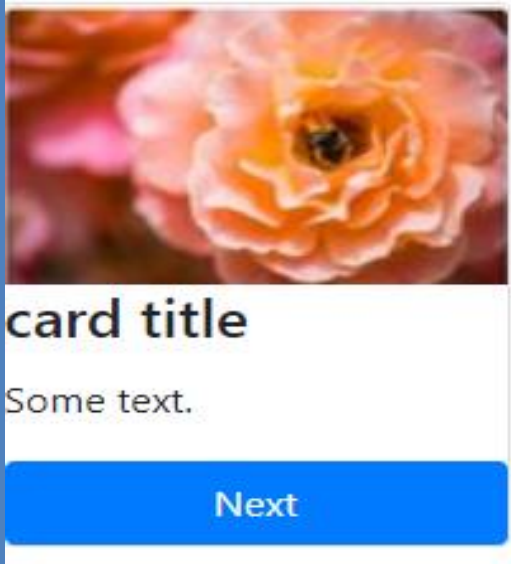
```
<div class="card">
  <div class="card-header">Header</div>
  <div class="card-body">Content</div>
  <div class="card-footer">Footer</div>
</div>
```

**.card-title:** add card titles to any heading element.

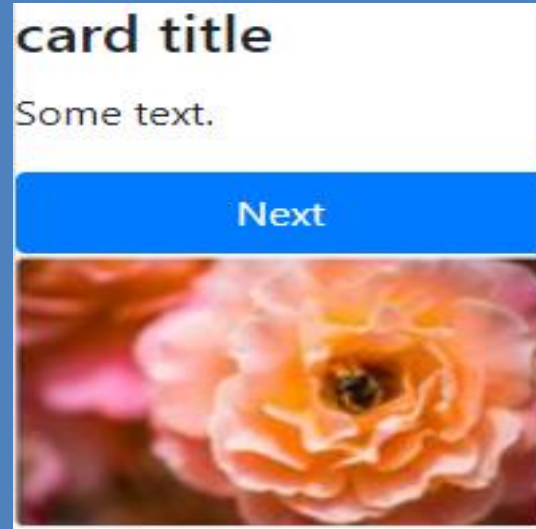
**.card-text:** used to remove bottom margins for a `<p>` element if it is the last child (or the only one) inside **.card-body**.

**.card-link** class adds a blue color to any link, and a hover effect.

`.card-img-top` to an `<img>` to place the image at the top inside the card.



`.card-img-bottom` to an `<img>` to place the image at the bottom inside the card.

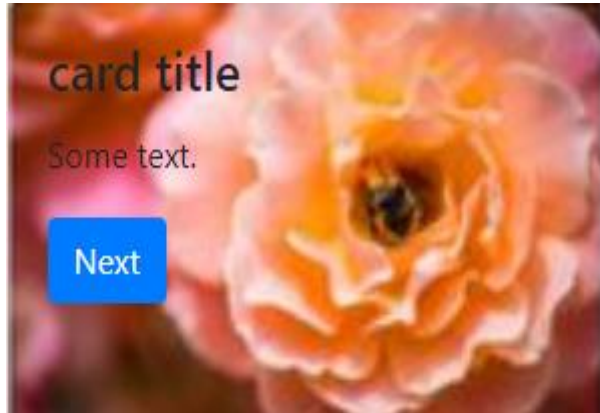


Apply `.stretched-link` class to a link inside the card, and it will make the whole card clickable and hoverable (the card will act as a link)

```
<a href="#" class="btn btn-primary stretched-link">See Profile</a>
```

## Card Image Overlays:

Turn an image into a card background and use `.card-img-overlay` to add text on top of the image



```
<div class="card" style="width:300px">
  
  <div class="card-img-overlay">
    <h4 class="card-title">John Doe</h4>
    <p class="card-text">Some text.</p>
    <a href="#" class="btn btn-primary">Next</a>
  </div>
</div>
```

`.card-columns` : creates a masonry-like grid of cards (like pinterest).

`.card-deck` class creates a grid of cards that are of **equal height and width**.

`.card-group` class is similar to `.card-deck`. The only difference is that the `.card-group` class removes left and right margins between each card.

The **collapse** is used to show and hide another element on the page.

**Ex.** `<button data-bs-toggle="collapse" data-bs-target="#col">Collapsible</button>`  
`<div id="col" class="collapse">`  
 This is the example of collapse....  
`</div>`

# Bootstrap Dropdown

For Dropdown we need to include popper.js file:

```
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"></script>
```

Class / Attribute	Purpose
dropdown	Specifies a dropdown menu
dropdown-toggle	Specifies the trigger element i.e the element which shows or hides the dropdown menu. In this example the trigger element is the button
data-toggle="dropdown"	This attribute is required on the trigger element to show or hide the dropdown menu
dropdown-menu	The unordered list with this class specifies the dropdown menu items

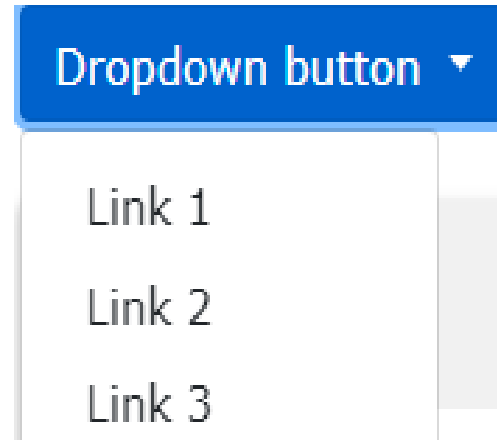
The `.divider` class is used to separate links inside the dropdown menu with a thin horizontal border.

```
<li> <hr class="dropdown-divider"></li>
```

The `.dropdown-header` class is used to add headers inside the dropdown menu.

```
<div class="dropdown-header">Dropdown header 1</div>
```

```
<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>
```



**Dropdown Position:** `.dropright` or `.dropleft`

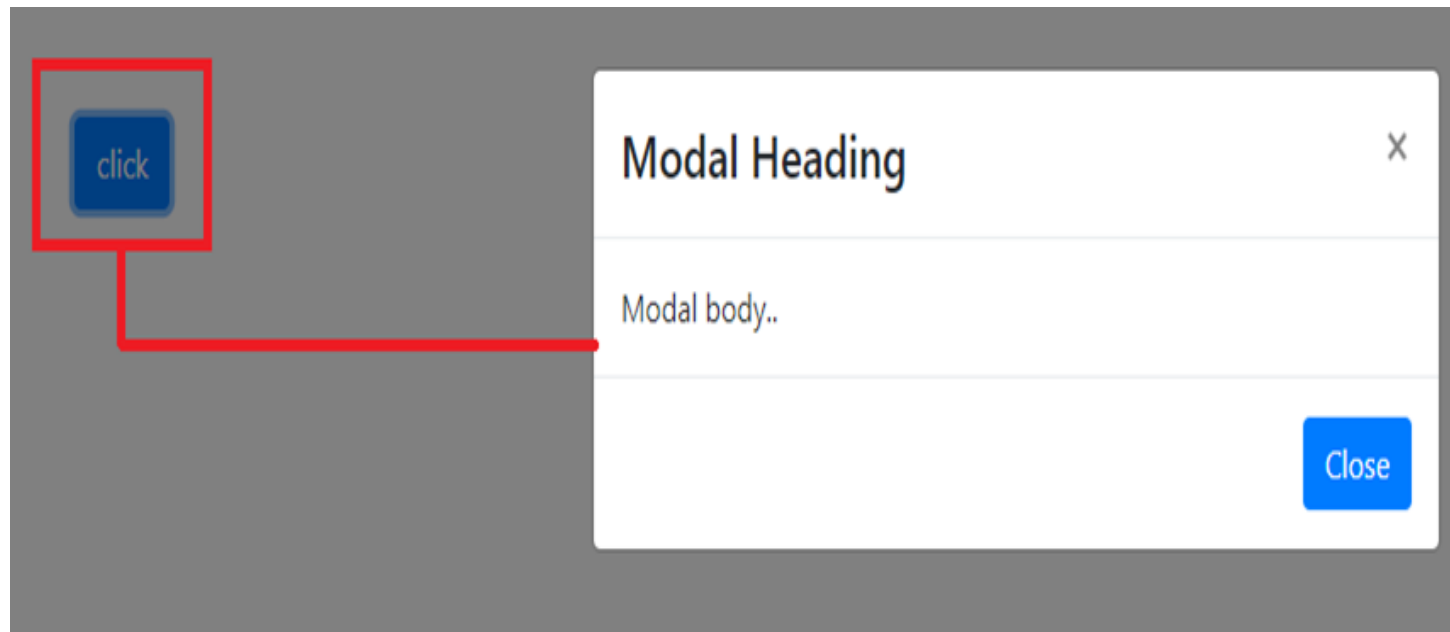
```
<div class="btn-group dropend">
  <div class="btn-group dropstart">
```

**Dropup:** If you want the dropdown menu to expand upwards instead of downwards.

```
<div class="btn-group dropup">
```

# Bootstrap modal popup

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



## Attributes and classes to customize the behaviour of the modal:

Class	Description
<code>.fade</code>	Use to add a fading effect when opening and closing the modal
<code>.modal-sm</code>	Creates a small modal.
<code>.modal-lg</code>	Creates a large modal.
<code>.modal-dialog-centered</code>	Center the modal vertically and horizontally within the page

Attribute	Description
<code>data-bs-backdrop="static"</code>	prevents modal closing when clicked outside the modal



```

<button type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#myModal">
  Open modal
</button>
<div class="modal" id="myModal">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-title">Modal Heading</h4>
        <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
      </div>
      <div class="modal-body">
        Modal body..
      </div>
      <div class="modal-footer">
        <button type="button" class="btn-close btn btn-primary" data-bs-
dismiss="modal">Close</button>
      </div>
    </div>
  </div>
</div>
</div>

```

# END

# Session-7

# Topics to be covered.....

- Bootstrap Forms
- Input Group
- Bootstrap Utility/Helper Classes
- Bootstrap Navs
- Navbar
- Carousel

**Bootstrap provides two types of form layouts:**

- **Stacked (full-width) form**
- **form**

The following are the classes that are used to style forms:

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

**.form-control** - Use this class on all textual elements (<input>, <textarea>, and <select>)

# Bootstrap Stacked Form

Email:

Password:

☐ Remember me

Submit

```
<form>
  <div class="form-group">
    <label for="email">Email:</label>
    <input type="email" class="form-control">
  </div>
  <div class="form-group">
    <label for="pwd">Password:</label>
    <input type="password" class="form-control">
  </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>
```

# Horizontal form

Email:

Password:

☐ Remember  
Me

Submit

```
<form>
  <div class="form-group row">
    <label for="email" class="col-form-label col-sm-2">Email:</label>
    <div class="col-sm-10">
      <input type="email" class="form-control" id="email">
    </div>
  </div>
  <div class="form-group row">
    <label for="pwd" class="col-form-label col-sm-2">Password:</label>
    <div class="col-sm-10">
      <input type="password" class="form-control" id="pwd">
    </div>
  </div>
  <div class="form-group row">
    <div class="col-sm-2 offset-sm-2">
      <div class="form-check">
        <input class="form-check-input" type="checkbox">
        <label class="form-check-label">
          Remember Me
        </label>
      </div>
    </div>
  </div>
  <button type="submit" class="btn btn-primary offset-sm-2">Submit</button>
</form>
```

## Bootstrap Textarea:

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

## Bootstrap Checkboxes:

```
< <div class="form-check"> <input class="form-check-input" type="checkbox" value=""
id="flexCheckDefault"> <label class="form-check-label" for="flexCheckDefault"> Default
checkbox </label> </div>
```


Use the **.form-check-inline** class if you want the checkboxes to appear on the same line



## Bootstrap Radio Buttons:

```
<div class="form-check"> <input class="form-check-input" type="radio"
name="flexRadioDefault" id="flexRadioDefault1"> <label class="form-check-label"
for="flexRadioDefault1"> Default radio </label> </div>
```

```
<input class="form-control" type="file" id="formFile">
```

A screenshot of a Bootstrap file input field. It consists of a light gray button with the text "Choose File" and a text input field containing the placeholder text "No file chosen".

## Bootstrap Select List

```
<select class="form-select" >
<option selected>Open this select menu</option>
<option value="1">One</option>
<option value="2">Two</option>
<option value="3">Three</option>
</select>
```

## Switches



```
<div class="form-check form-switch">
  <input class="form-check-input" type="checkbox">
  <label class="form-check-label">Default switch checkbox input</label>
</div>
```

## Floating labels:-

```
<div class="form-floating mb-3">
  <input type="email" class="form-control" id="floatingInput" placeholder="name@example.com">
  <label for="floatingInput">Email address</label>
</div>
```

# Inputs Group

Class	Description
<code>.input-group</code>	A container to enhance an input by adding an icon, text or a button in front or behind the input field as a "help text".
<code>.input-group-prepend</code>	to add the help text in front of the input
<code>.input-group-append</code>	to add it behind the input.
<code>.input-group-text</code>	style the specified help text

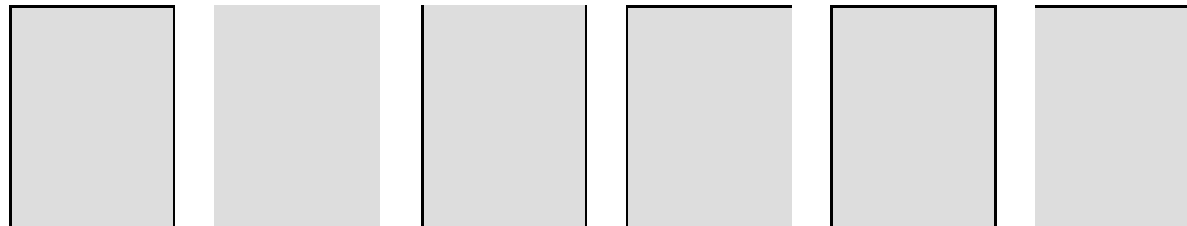
```
<div class="input-group">
  <span class="input-group-text">$</span>
  <input type="text" class="form-control">
</div>
```



Bootstrap 4 has a lot of utility/helper classes to quickly style elements without using any CSS code.

## Borders:

```
<span class="border"></span>
<span class="border border-0"></span>
<span class="border border-top-0"></span>
<span class="border border-right-0"></span>
<span class="border border-bottom-0"></span>
<span class="border border-left-0"></span>
```



## Border color:

```
<span class="border border-primary"></span>
<span class="border border-secondary"></span>
<span class="border border-success"></span>
<span class="border border-danger"></span>
<span class="border border-warning"></span>
<span class="border border-info"></span>
<span class="border border-light"></span>
<span class="border border-dark"></span>
<span class="border border-white"></span>
```



## Border Radius:



.rounded, .rounded-top, .rounded-right, .rounded-bottom, .rounded-left, rounded-circle, .rounded-0

## Width & Height:

.w-25 : Width 25%

.w-50 : Width 50%

.w-75 : Width 75%

.w-100 : Width 100%

.mw-100 : Max Width 100%

.h-25 : Height 25%

.h-50 : Height 50%

.h-75 : Height 75%

.h-100 : Height 100%

.mh-100 : Max Height 100%

# Spacing:

Bootstrap 4 has a wide range of responsive margin and padding utility classes.

The classes are named using the format :

`{property}{sides}-{size}` for `xs` and

`{property}{sides}-{breakpoint}-{size}` for `sm`, `md`, `lg`, and `xl`.

Where *property* is one of:

`m` - for classes that set `margin`

`p` - for classes that set `padding`

## Where *sides* is one of:

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

## Where *size* is one of:

- **0** - for classes that eliminate the **margin** or **padding** by setting it to **0**
- **1** - (by default) for classes that set the **margin** or **padding** to **\$spacer \* .25**
- **2** - (by default) for classes that set the **margin** or **padding** to **\$spacer \* .5**
- **3** - (by default) for classes that set the **margin** or **padding** to **\$spacer**
- **4** - (by default) for classes that set the **margin** or **padding** to **\$spacer \* 1.5**
- **5** - (by default) for classes that set the **margin** or **padding** to **\$spacer \* 3**
- **auto** - for classes that set the **margin** to **auto**



## Display property:

To make an element into a block element.

As such, the classes are named using the format:

- `.d-{value}` for `xs`
- `.d-{breakpoint}-{value}` for `sm`, `md`, `lg`, and `xl`.

```
<span class="d-block bg-success">d-block</span>
```

```
<span class="d-sm-block bg-success">d-sm-block</span>
```

```
<div class="d-inline p-2 bg-primary text-white">d-inline</div>
```

```
<div class="d-inline p-2 bg-dark text-white">d-inline</div>
```

d-inline

d-inline

## Nav Menus:

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

Home      About Us

## Aligned Nav:

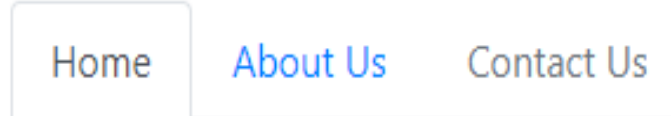
- .justify-content-center** class to center the nav.
- .justify-content-end** class to right-align the nav.

## Vertical Nav: `<ul class="nav flex-column">`

Home

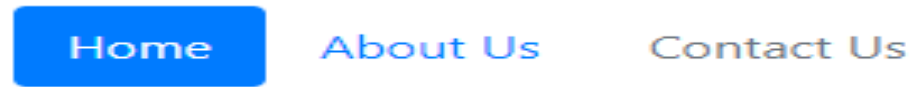
About Us

## Tabs:



```
<ul class="nav nav-tabs">
  <li class="nav-item">
    <a class="nav-link
active" href="#">Home</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">About
Us</a>
  </li>
  <li class="nav-item">
    <a class="nav-link
disabled" href="#">Contact Us</a>
  </li>
</ul>
```

**Pills:** `<ul class="nav nav-pills">`



`.nav-justified`: Justify the tabs/pills.

**Fill and justify:**

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

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

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

  <ul class="navbar-nav">
    <li class="nav-item active">
      <a class="nav-link" href="#">Home</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">About</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Contact</a>
    </li>
  </ul>

</nav>
```



## Vertical Navbar:

Remove the `.navbar-expand-xl|lg|md|sm` class to create a vertical navigation bar.

Class	Description
<code>.justify-content-center</code>	center the navigation bar
<code>.navbar-brand</code>	used to highlight the brand/logo/project name of your page
<code>.fixed-top</code>	makes the navigation bar fixed at the <b>top</b>
<code>.fixed-bottom</code>	make the navbar stay at the <b>bottom</b> of the page
<code>.sticky-top</code>	make the navbar fixed/stay at the <b>top</b> of the page when you scroll <b>past</b> it
<code>.navbar-dark</code>	Add a <b>white</b> text color to all links in the navbar
<code>.navbar-light</code>	add a <b>black</b> text color to all links in the navbar
<code>.navbar-text</code>	Use to vertical align any elements inside the navbar that are not links

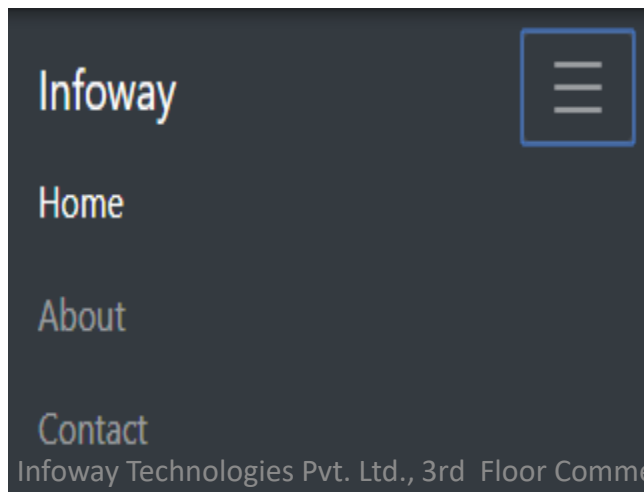
# Bootstrap navbar component

It is useful for creating responsive navigation header for a website. When the toggle button is clicked the navbar expands. However, on large screen devices like laptops and desktops, where we have enough room to display the entire navbar, it appears normal.

**Creating responsive navbar :** The navbar appears normal on a large screen size devices like laptops, desktops etc.



The navbar appears collapsed on a small screen size devices like mobile phones, portrait tablets etc. The navbar expands when the toggle button is clicked.



```
<nav class="navbar navbar-expand-md bg-dark navbar-dark">
  <!-- Brand -->
  <a class="navbar-brand" href="#">Infoway</a>

  <!-- Toggler/collapsible Button -->
  <button class="navbar-toggler" type="button" data-bs-
toggle="collapse" data-bs-target="#colNav">
    <span class="navbar-toggler-icon"></span>
  </button>

  <!-- Navbar links -->
  <div class="collapse navbar-collapse" id="colNav">
    <ul class="navbar-nav">
      <li class="nav-item active">
        <a class="nav-link" href="#">Home</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">About</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Contact</a>
      </li>
    </ul>
  </div>
</nav>
```



# Bootstrap Carousel Plugin

The Carousel is a slideshow for cycling through elements.



```
<div id="myCarousel" class="carousel slide" data-bs-ride="carousel">
```

Class	Description
<code>.carousel</code>	Creates a carousel
<code>.carousel-indicators</code>	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)
<code>.carousel-inner</code>	Adds slides to the carousel
<code>.carousel-item</code>	Specifies the content of each slide
<code>.carousel-control-prev</code>	Adds a left (previous) button to the carousel, which allows the user to go back between the slides
<code>.carousel-control-next</code>	Adds a right (next) button to the carousel, which allows the user to go forward between the slides
<code>.carousel-control-prev-icon</code>	Used together with <code>.carousel-control-prev</code> to create a "previous" button
<code>.carousel-control-next-icon</code>	Used together with <code>.carousel-control-next</code> to create a "next" button
<code>.slide</code>	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

# END

# Thank You