

Full Stack



Caltech

Center for Technology &
Management Education

Post Graduate Program in Full Stack Web Development

Full Stack



Caltech

**Center for Technology &
Management Education**

**Develop a Web Application using
frontend stack**



Customize Your Application Using Bootstrap

A Day in the Life of a Full Stack Developer

In this sprint, Joe has to develop a customizable website for a restaurant that will provide a brief information on the restaurant, its bestsellers, and its contact number.

In this lesson, you will learn how to solve this real-world scenario to help Joe complete his task effectively and quickly.

Learning Objectives

By the end of this lesson, you will be able to:

- 👁 Explain HTML, CSS, Bootstrap typography and utilities
- 👁 Illustrate the Bootstrap design
- 👁 List buttons, navigation bars, input groups, and media objects
- 👁 Describe carousel, accordion, modal, and pagination
- 👁 Define forms, tabs, and captions



Starter Template Guide

You Already Know

Course(s):

1. A Front-End Web Developer MasterClass Using HTML, CSS, and JavaScript
2. Build Real World Websites from Scratch Using HTML5 and CSS3



- Apply designs of HTML5 and CSS3
 - Basics of HTML and CSS
 - What we'll create in this course?
- Build your own HTML5 video player
 - Build HTML5 video player
- Describe HTML5 input types
 - HTML5 input types



- Get your free Web Hosting: HTML
 - Basics of HTML and CSS Structure of a website
 - Heading, Paragraphs, Links, Images, and Tables
- Explain the fundamentals of CSS
 - Introduction
 - CSS selectors
 - CSS transforms
- Create a project using HTML and CSS
 - Headers
 - Menu
 - Sidebars



A Day in the Life of a Full Stack Developer

Joe has gained expertise in Bootstrap. A new project for the food industry is assigned to him.

In this sprint, he has to develop a customizable website for a restaurant that will provide a brief information on the restaurant, its bestsellers, and its contact number.

In this lesson, you will learn how to solve this real-world scenario to help Joe complete his task effectively and quickly.

Responsive Web Design: Introduction

Responsive Web Design is a responsive browser that can adapt to different screen sizes.



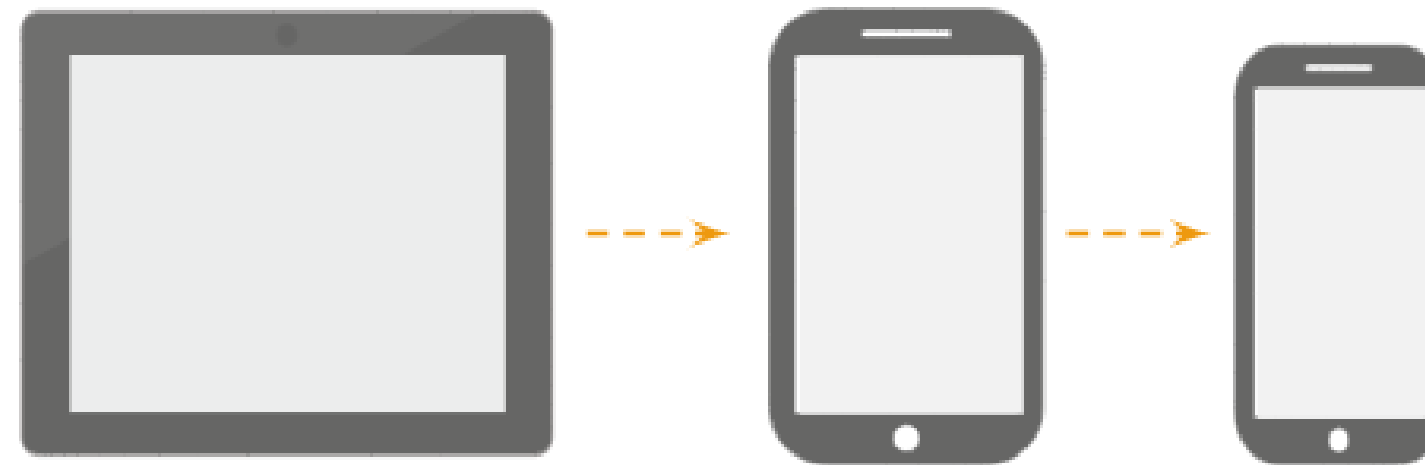
Responsive vs. Adaptive

Responsive

Responsive web design is a fluid design that responds to the browser and can resize according to the screen size with no empty spaces on the sides.

Adaptive

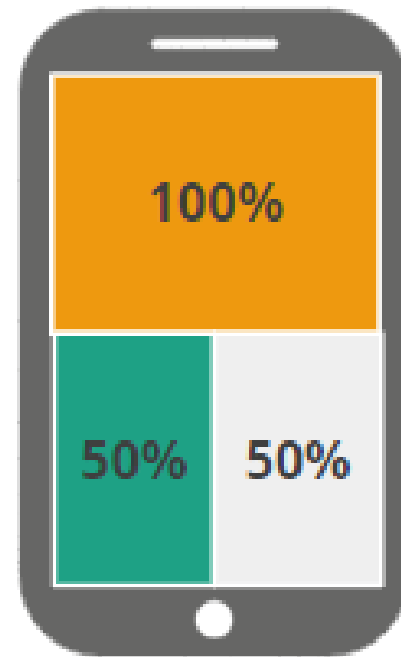
Adaptive design adapts to the screen size only at specific points.



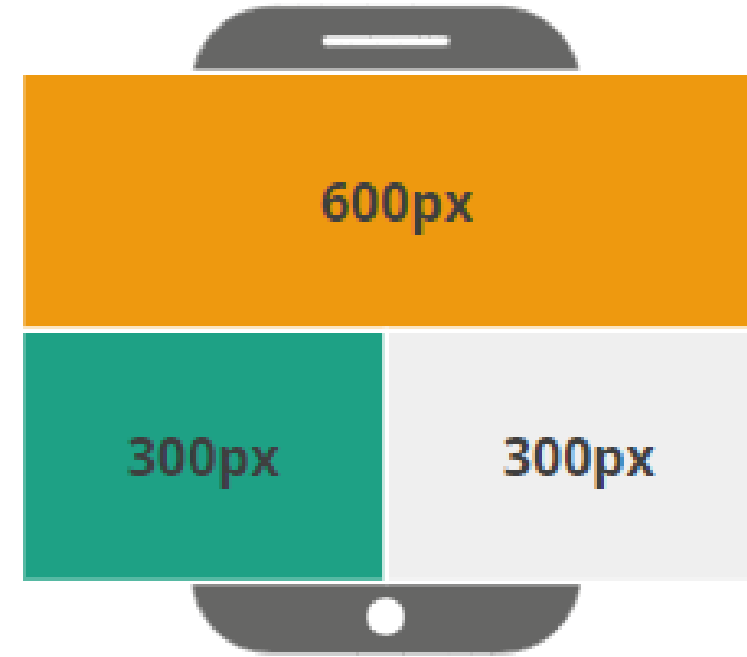
Responsive Web Design

Relative vs. Static Units

Relative units can adapt to any screen in percentages, unlike static units that adapt in pixels.



Relative units



Static units

Responsive Web Design: Introduction

The flow in responsive web design helps the content to smoothly adjust when the screen size changes, unlike in static web design.



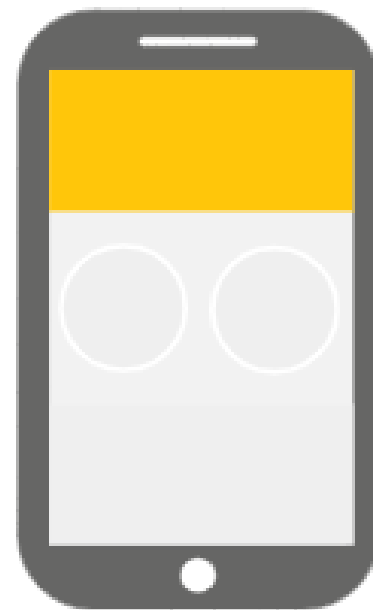
Flow



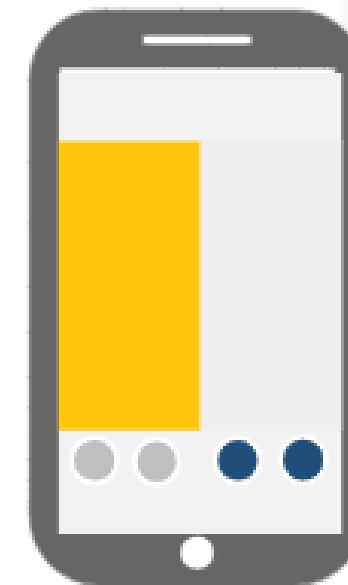
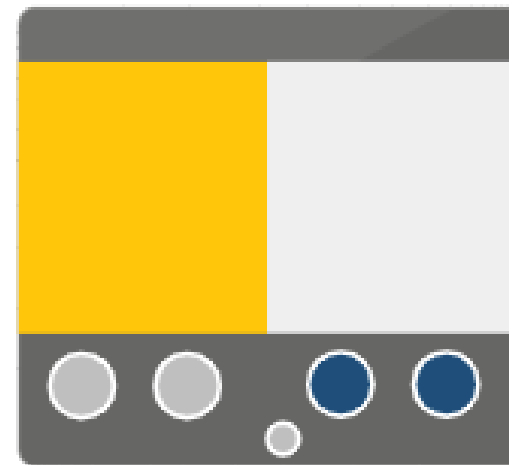
Static

With vs. Without Breakpoints

When the screen size changes, the presence of breakpoints allows the content to adapt to the screen size.



With Breakpoints



Without Breakpoints

Vectors vs. Images

A vector adapts to the resolution of the retina, and its quality is independent of the screen resolution. In the case of an image, it cannot adapt to the resolution of the retina.

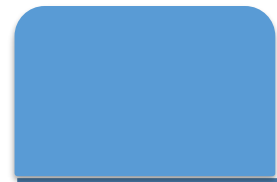


Vector



Image

Mobile First



To develop a mobile-first design:

- Start with CSS development for mobile device, followed by media queries to adapt to desktops and tablets
- Determine the content that is the most important



Why Mobile First?

Is a design that is independent of the screen size

Helps avoid overwriting of code

Narrows down the website information to a convenient portion



Properties of Bootstrap



Preprocessors



**One framework for
every device**



Customizable

Features of Bootstrap



- Easy to Use
- Mobile-Friendly
- Browser Compatibility
- Powerful Grid system
- Bundled JavaScript Plugins
- Mobile-first approach
- Simple Integration

Competitors

These are a few front-end web designing alternatives that can be used instead of Bootstrap:



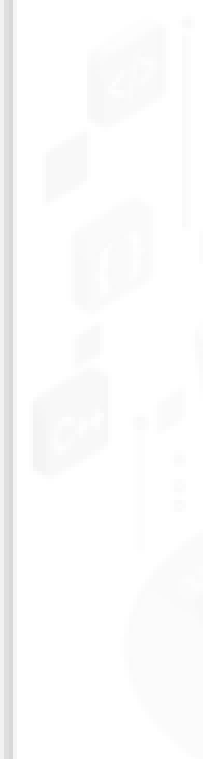
Foundation



Skeleton

Bootstrap Components

- Labels
- Panels
- Jumbotron
- Page Header
- Navbar
- Thumbnails
- Alerts
- List group
- Page Header
- Wells
- Badges
- Dropdowns
- Button groups
- Glyphicons
- Button Drops
- Breadcrumbs
- Input groups
- Navs
- Pagination
- Media Object



Creating First Page with Bootstrap: Step 1

At the start of the page, include HTML5 doctype and the lang attribute with the right character set.

```
<!DOCTYPE html>
<html lang="pt">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1">
  </head>
</html>
```

Creating First Page with Bootstrap: Step 2

- Add the viewport <meta> tag for proper touch zooming and rendering
- Adjust the width and the initial zoom level of the page as per your device
- This will ensure that you have a better experience and there will be no need for zooming

Creating First Page with Bootstrap: Step 3

- Include the following files:
 - bootstrap.min.css
 - jquery.min.js
 - bootstrap.min.js
- Download and include the files in your page. In this example, we include them from the Internet:

```
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
<script
src="https://maxcdn.bootstrapcdn.com/ajax/bootstrap/3.3./js/bootstrap.min.js"></script>
```

Creating First Page with Bootstrap: Step 4

- In Bootstrap, you can include elements to wrap the site contents.

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

Assisted Practice

Import Bootstrap in Your Application

Duration: 20 Min.

Problem Statement:

You are given a project to create an HTML page and include Bootstrap to your web application.

Assisted Practice: Guidelines

Steps to import bootstrap in your application:

1. Write an HTML page in your code editor
2. Add bootstrap to your application using CDN
3. Host bootstrap locally
4. Push the code to GitHub repositories

Browsers and Devices Compatibility

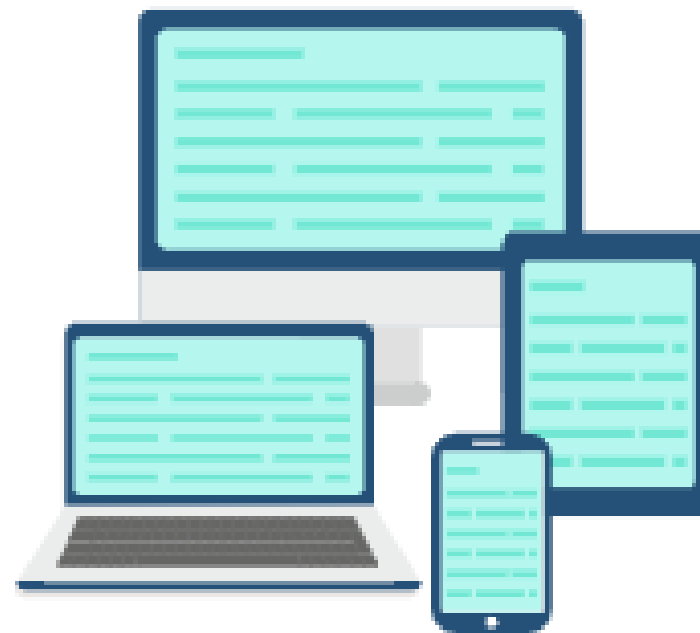
Responsive with Bootstrap

Bootstrap helps you to create responsive and mobile-friendly websites.

It offers:

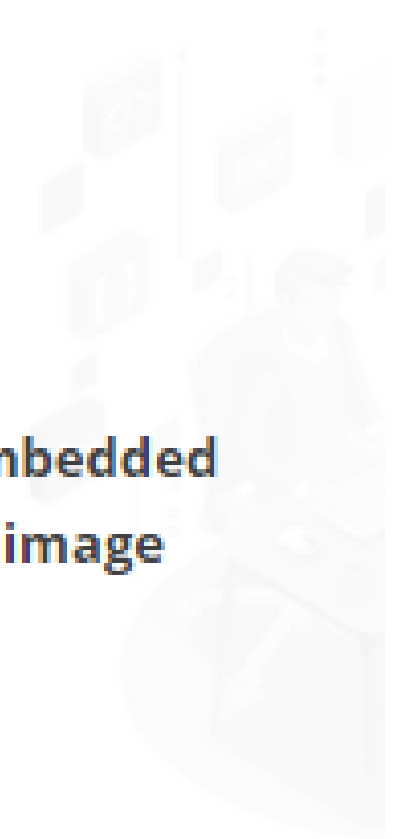
Grid system

CSS3 media queries



**Responsive embedded
and flexible image**

Responsive utilities



Responsive with Bootstrap

With @media rule in Bootstrap, you can have a different layout for all screens, including tablet, mobile, and desktops.



Introduction to Grid Systems



In Bootstrap

- Grid system permits a maximum of 12 columns on a page.
- You can group and create wider columns instead of using them individually.
- For decent organization and padding, place the rows within `.container-fluid` (full-width) or `.container` (fixed-width).
- Group columns horizontally using rows.
- Create grid columns by specifying the number of available columns you wish to span.
- For example, three equal columns would use three `.col-xs-4`.
- Place the content within columns, and only columns may be immediate children of rows.

Grid System

In Bootstrap, the grid system presents a swift and simple way to build responsive website layouts.

	Extra small devices Phones (<768px)	Small devices Tablets (≥768px)	Medium devices Desktops (≥992px)	Large devices Desktops (≥1200px)
Grid behavior	Horizontal at all times	Collapsed to start, horizontal above breakpoints		
Container width	None (auto)	750px	970px	1170px
Class prefix	.col-xs-	.col-sm-	.col-md-	.col-lg-
# of columns	12			
Column width	Auto	~62px	~81px	~97px
Gutter width	30px (15px on each side of a column)			

With vs. Without Breakpoints

The Bootstrap grid system has four classes:



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

Bootstrap: Cards, Tables, and Lists

Typography




Typography is a Bootstrap feature for formatting and styling text content.

Headings are set by HTML tags from <H1> to <H6>, and some others such as <mark>, <small>, and .

HTML code	Display
<p class="h1">Heading H1</p>	Heading H1
<p class="h2">Heading H2</p>	Heading H2
<p class="h3">Heading H3</p>	Heading H3
<p class="h4">Heading H4</p>	Heading H4
<p class="h5">Heading H5</p>	Heading H5
<p class="h6">Heading H6</p>	Heading H6

Images

Image classes in Bootstrap are responsive so that they scale in size according to the size of their parent elements without losing the aspect ratio.

HTML code	Display
<code></code>	
<code></code>	
<code></code>	

Media Objects

Media Objects are components to display images or videos that are aligned to the left or right of the content.

Usually, blog comments and tweets are displayed as media objects.

Syntax:

```
<div class="media-object">
  <div class="media-object-section">
    <div class="thumbnail">
      
    </div>
  </div>
  <div class="media-object-section">
    <h4>Comment heading.</h4>
    <p>Some comments here.....</p>
  </div>
</div>
```

Display:



Comment heading.

Some comments here.....

Cards

Cards are used to display content made of different types of objects such as text or image.

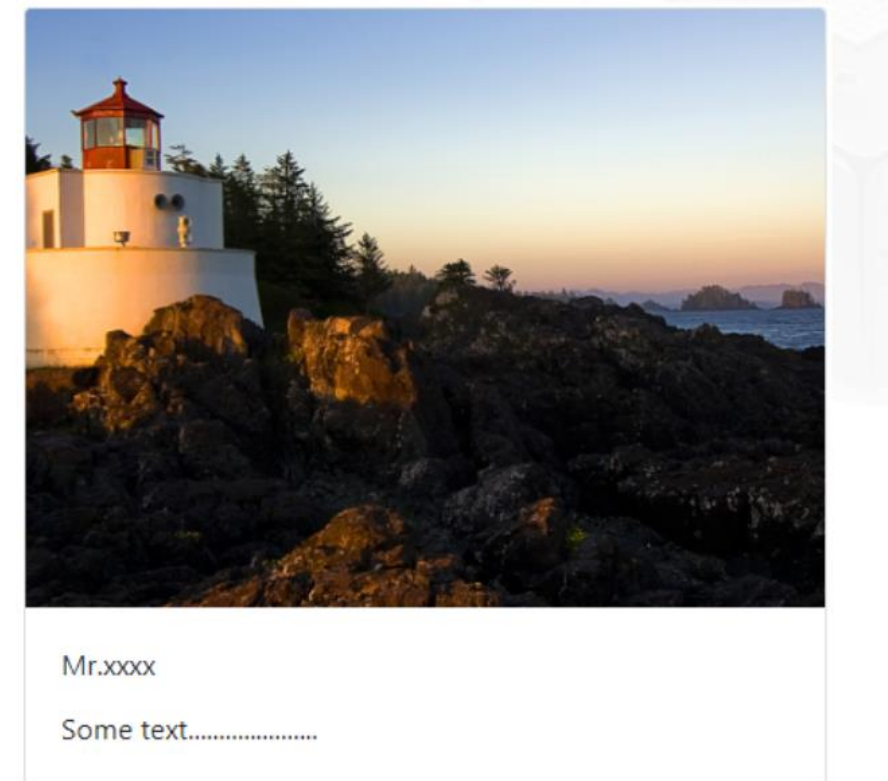
Basic card:

```
<div class="card">
  <div class="card-body">This is a basic
card</div>
</div>
```

This is a basic card.

Image card:

```
<div class="card" style="width:450px">
  
  <div class="card-body">
    <p class="card-title">Mr.xxxx</p>
    <p class="card-text">Some text...</p>
  </div>
</div>
```



Cards

A Jumbotron is a component to display a message in a grey box to call for extra attention to the text.

Syntax:

```
<div class="jumbotron">  
  <h1>Jumbotron</h1>  
  <p>It is a grey box to highlight a message.</p>  
</div>
```

Display:

Jumbotron

It is a grey box to highlight a message.

Badges








Badges, or labels, are used to highlight or show additional information related to a particular content.

HTML code	Display
<code><h1>Gifts Free</h1></code>	Gifts Free
<code><h2>Laptops New</h2></code>	Laptops New
<code><h3>Tablets New arrivals</h3></code>	Tablets New arrivals
<code><h4>Smart phones Coming soon</h4></code>	Smart phones Coming soon
<code><h5>Inbox 12</h5></code>	Inbox 12

Glyphicons

Glyphicons are icon fonts that can be used in web pages in the forms of text, toolbars, forms, buttons, and navigation toolbars.

Example:

HTML Code	Display
<code></code>	
<code></code>	
<code></code>	
<code></code>	
<code></code>	
<code></code>	
<code></code>	

Tables

Tables allow data such as text, images, or links to be arranged into rows and columns.

HTML code	Display												
<pre><table class="table table-striped table-bordered"> <thead> <tr> <th>Heading 1</th> <th>Heading 2</th> <th>Heading 3</th> </tr> </thead> <tbody> <tr> <td>Cell 11</td> <td>Cell 12</td> <td>Cell 13</td> </tr> <tr> <td>Cell 21</td> <td>Cell 22</td> <td>Cell 23</td> </tr> <tr> <td>Cell 31</td> <td>Cell 32</td> <td>Cell 33</td> </tr> </tbody> </table></pre>	<table><tr><th>Heading 1</th><th>Heading 2</th><th>Heading 3</th></tr><tr><td>Cell 11</td><td>Cell 12</td><td>Cell 13</td></tr><tr><td>Cell 21</td><td>Cell 22</td><td>Cell 23</td></tr><tr><td>Cell 31</td><td>Cell 32</td><td>Cell 33</td></tr></table>	Heading 1	Heading 2	Heading 3	Cell 11	Cell 12	Cell 13	Cell 21	Cell 22	Cell 23	Cell 31	Cell 32	Cell 33
Heading 1	Heading 2	Heading 3											
Cell 11	Cell 12	Cell 13											
Cell 21	Cell 22	Cell 23											
Cell 31	Cell 32	Cell 33											

List Groups


List groups are used to display a set of elements in an ordered or unordered list.

	HTML code	Display
Unordered List	<pre><ul class="list-group"> <li class="list-group-item">First <li class="list-group-item">Second <li class="list-group-item">Third <li class="list-group-item">Fourth <li class="list-group-item">Fifth </pre>	<div>First</div> <div>Second</div> <div>Third</div> <div>Fourth</div>
Ordered List	<pre> First Second Third Fourth </pre>	<div>1. First</div> <div>2. Second</div> <div>3. Third</div> <div>4. Fourth</div>

Buttons

A button is used to trigger an action.


Optional style and color classes can be used to change the appearance of a web page.

HTML code	Display
<pre><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></pre>	

Stateful Buttons

Stateful buttons can be used to toggle between states.

These buttons can change text and color when clicked or hovered over.

HTML Code	Display
<pre><label class="btn btn-primary active"> <input type="checkbox" name="options"> Option A </label> <label class="btn btn-primary"> <input type="checkbox" name="options"> Option B </label> <label class="btn btn-primary active"> <input type="checkbox" name="options"> Option C </label></pre>	

Header and Footer

Bootstrap **header** is a method used to hold elements. It is usually placed above the body element.

```
<div class="card-header bg-transparent  
border-success">Bootstrap</div>
```

Bootstrap **footer** is a navigation method used to hold links, buttons, and other elements. It is usually placed at the end of a web page.

```
<div class="card-footer bg-transparent border-success">Updated 3 minutes ago</div>
```

Header and Footer

```
<div class="card border-success mb-3" style="max-width: 18rem;">
```

```
<div class="card-header bg-transparent border-success">Bootstrap</div>
```

```
<div class="card-body text-success">  
  <h5 class="card-title">Header and Footer</h5>  
  <p class="card-text">This is the simplest way of  
using header and footer</p>  
</div>
```

```
<div class="card-footer bg-transparent border-success">Updated 3 minutes ago</div>  
</div>
```

Output

Bootstrap

Header and Footer

This is the simplest way of using header and footer

Updated 3 minutes ago

Text Alignments: Text-Left

Bootstrap allows the alignment of text to the left side of an HTML page.

```
<div class="card" style="width: 18rem;">
  <div class="card-body">
    <h5 class="card-title">Rock Music</h5>
    <p class="card-text">A popular genre of
music.</p>
    <a href="#" class="btn btn-
primary">Buy</a>
  </div>
</div>
```

Output

**Rock
Music**
A popular genre of music.

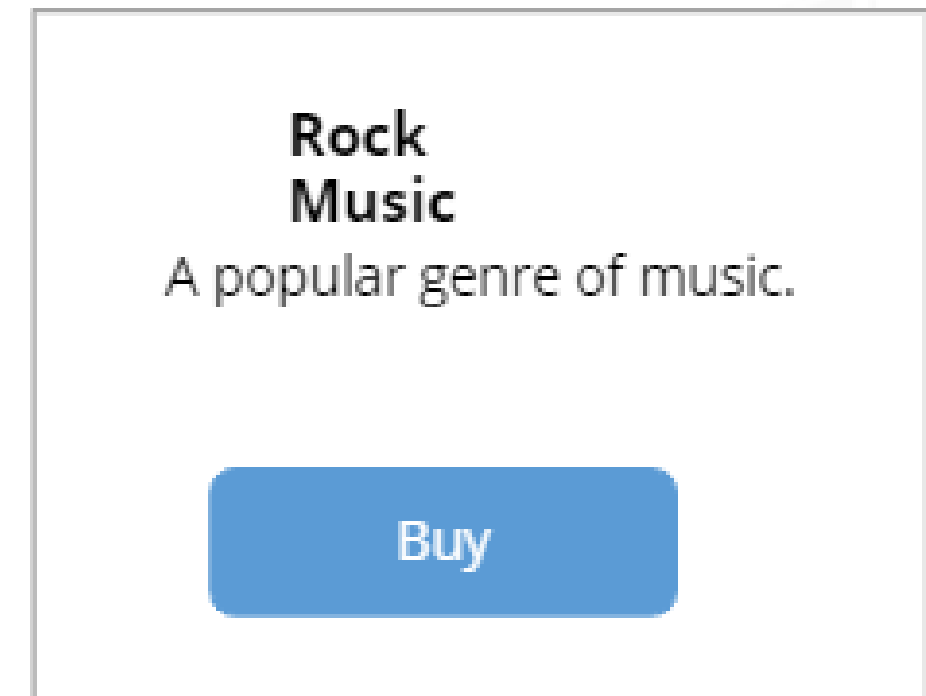
Buy

Text Alignments: Text-Center

Bootstrap allows the alignment of the text to the center of an HTML page.

Output

```
<div class="card text-center" style="width:
18rem;">
  <div class="card-body">
    <h5 class="card-title">Rock Music</h5>
    <p class="card-text">A popular genre of
music.</p>
    <a href="#" class="btn btn-
primary">Buy</a>
  </div>
</div>
```

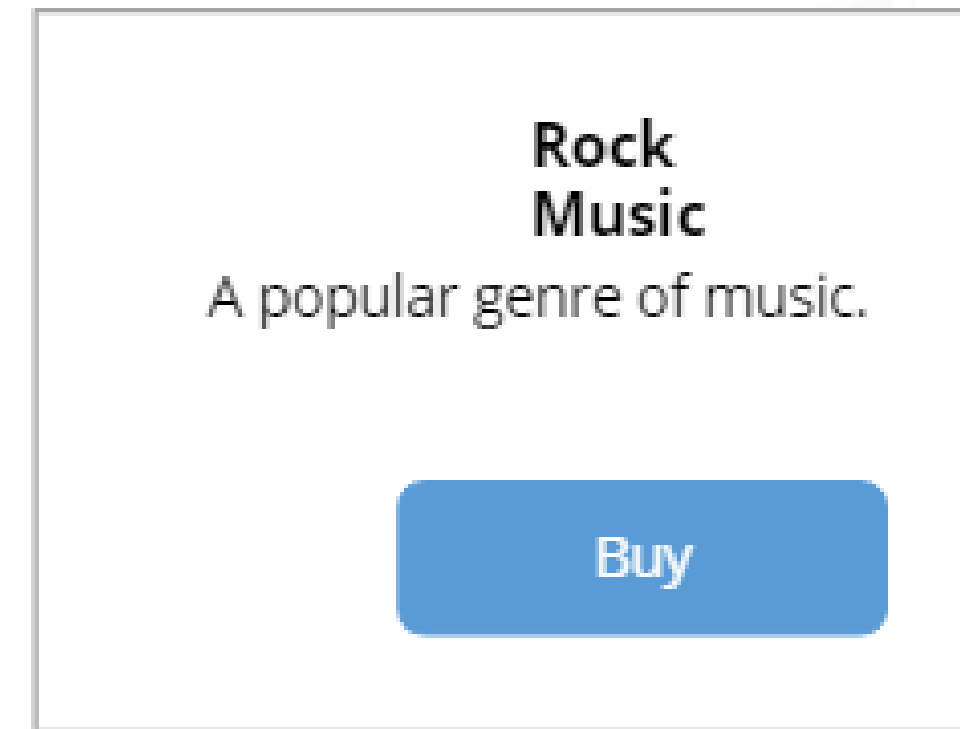


Text Alignments: Text-Right

Bootstrap allows the alignment of the text to the right side of an HTML page.

Output

```
<div class="card text-right" style="width:
18rem;">
  <div class="card-body">
    <h5 class="card-title">Rock Music</h5>
    <p class="card-text">A popular genre of
music.</p>
    <a href="#" class="btn btn-
primary">Buy</a>
  </div>
</div>
```



Assisted Practice

Develop a Static Web Page

Duration: 50 Min.

Problem Statement:

You are given a project to develop a static web page.

Assisted Practice: Guidelines

Steps to develop a static web page:

1. Build a static web page in your code editor
2. View the results in your local browser
3. Push the code to GitHub repositories

Bootstrap: Forms and Form Validations

Input Group

Input groups are extended form controls which are used to add text or buttons before or after text-based inputs.

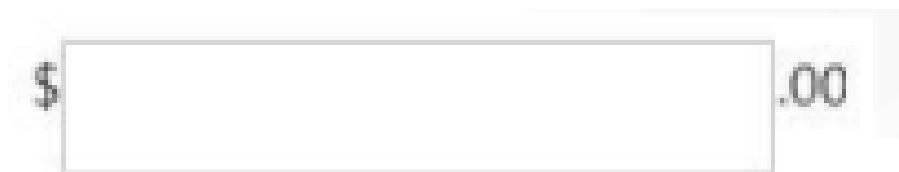
Syntax:

```
<div class = "input-group">  
  <input type = "text" class = "form-control"  
placeholder = "login">  
  <span class = "input-group-addon">@</span>  
  <input type = "text" class = "form-control"  
placeholder = "xxx.com">  
</div>
```

Display:



```
<div class = "input-group">  
  <span class = "input-group-addon">$</span>  
  <input type = "text" class = "form-  
control">  
  <span class = "input-group-  
addon">.00</span>  
</div>
```



Forms

Forms are elements used to collect data from visitors to the website.

They can be text boxes, buttons, checkboxes, radio buttons, login fields, and password fields.

HTML Code	Display
<pre><form action="#"> <div class="form-group"> <label>Enter Email</label> <input type="email" class="form-control" placeholder="mailid@xxx.yyy"> </div> <div class="form-group"> <label>Password</label> <input type="password" class="form-control" placeholder="Password"> </div> <button type="submit">Login</button> </form></pre>	

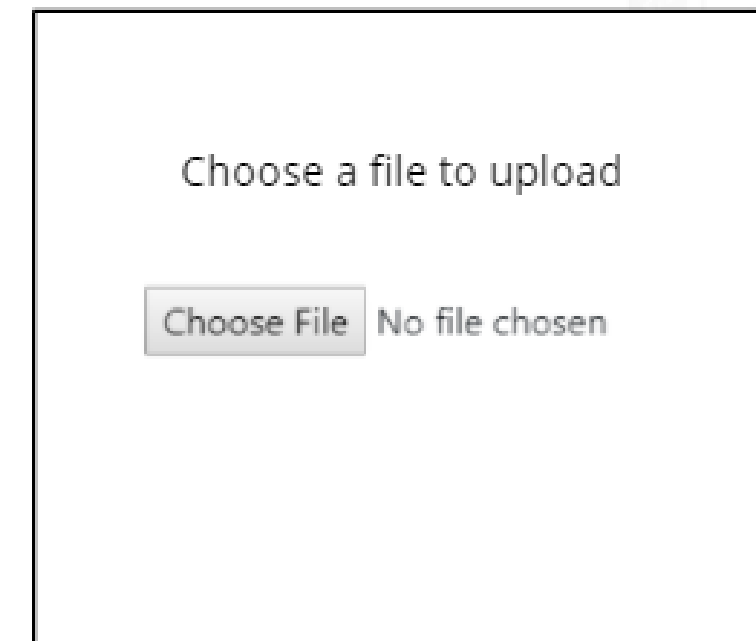
Form Controls

Form controls such as `<select>` and `<input>` are styled using the `.form-control` class.

Using *form-control-file*:

```
<form>
  <div class="form-group">
    <label for="formcontrol">Choose a
    file to upload</label>
    <input type="file" class="form-
    control-file" id="formcontrol">
  </div>
</form>
```

Output:

The screenshot shows a web form with a label "Choose a file to upload" and a file input control. The input control consists of a button labeled "Choose File" and a text area displaying "No file chosen". The entire form is enclosed in a black rectangular border.

Custom Forms

Custom forms are HTML elements that replace default elements of a browser to allow customization and consistency across browsers.

HTML Code	Display
<pre><h6>Age Group</h6> <div class="custom-control custom-radio"> <input type="radio" class="custom-control-input"> <label class="custom-control-label">20 to 50</label> </div> <div class="custom-control custom-radio"> <input type="radio" class="custom-control-input"> <label class="custom-control-label">Above 50</label> </div></pre>	<div>Age Group</div> <div><div></div> 20 to 50</div> <div><div></div> Above 50</div>

Form Controls: Sizing

Form control sizes can be set using `.form-control-lg` and `.form-control-sm`.

Using *form-control-lg* and *form-control-sm*:

```
<select class="form-control
form-control-lg">
  <option>Large select</option>
</select>
<select class="form-control">
  <option>Default
select</option>
</select>
<select class="form-control
form-control-sm">
  <option>Small select</option>
</select>
```

Output:

The image shows three vertically stacked dropdown menus. The top menu is the largest and is labeled 'Large select'. The middle menu is of standard size and is labeled 'Default select'. The bottom menu is the smallest and is labeled 'Small select'. Each menu has a downward-pointing arrow on its right side.

Form Controls: Inline

Checkboxes and radios can be placed horizontally by adding `.form-check-inline` to any `.form-check`.

Using *form-control-inline*:

```
<div class="form-check form-check-inline">
  <input class="form-check-input"
  type="checkbox" id="inlineCheckbox1"
  value="option1">
  <label class="form-check-label"
  for="inlineCheckbox1">1</label>
</div>
<div class="form-check form-check-inline">
  <input class="form-check-input"
  type="checkbox" id="inlineCheckbox2"
  value="option2">
  <label class="form-check-label"
  for="inlineCheckbox2">2</label>
</div>
```

Output:



Form Controls: Grid

Form grids are used to build complex forms.

Using *row*:

```
<form>
  <div class="row">
    <div class="col">
      <input type="text" class="form-control" placeholder="First name">
    </div>
    <div class="col">
      <input type="text" class="form-control" placeholder="Last name">
    </div>
  </div>
</form>
```

Output:

The output shows a rectangular container with a thin black border. Inside, there are two text input fields arranged horizontally. The first field on the left has a light blue border and contains the placeholder text "First name". The second field on the right also has a light blue border and contains the placeholder text "Last name".

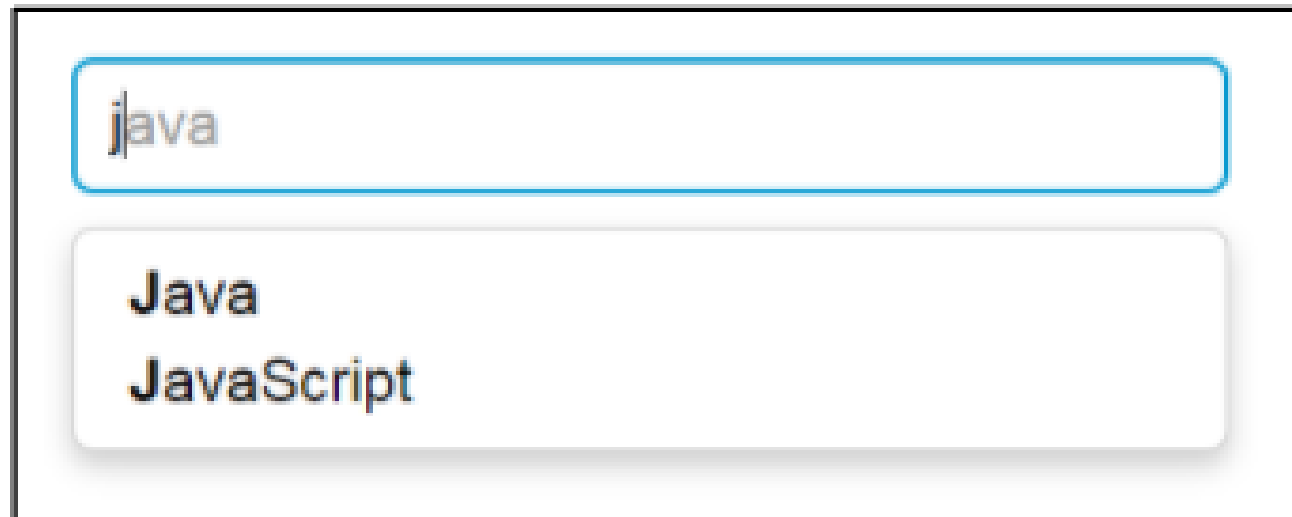
Typeaheads

Typeaheads are used to show hints to the users while filling forms for auto-completing them. This saves time and reduces input mistakes.

Syntax:

```
<script type="text/javascript">
$(document).ready(function(){
var cars = ['HTML', 'HTML5', 'Java', 'JavaScript', 'CSS', 'DTD', 'Bootstrap'];
</script>
```

Output:



Typeaheads

Forms can be disabled from user interactions using the disabled boolean attribute. It can be used to a *<fieldset>* to disable form controls.

```
<form>
  <fieldset disabled>
<div class="form-group">
  <label for="disabledSelect">Disabled select menu</label>
  <select id="disabledSelect" class="form-control">
    <option>Disabled select</option>
  </select>
</div>
<button type="submit" class="btn btn-primary">Submit</button>
</fieldset>
</form>
```

Forms Validation

Forms validation is a method by which the server checks if a user has entered all the necessary data in the required format.

Functions of form validation:

Basic Validation

A form is checked for any missing data in mandatory fields

Data Form Validation

A form is checked for data accuracy

Tabs

Tabs are components in which content or links are placed in separate panes to open different pages and sections.

Syntax:

```
<ul class="nav nav-tabs">  
  <li class="active"><a href="#">Home</a></li>  
  <li><a href="#">Menu 1</a></li>  
  <li><a href="#">Menu 2</a></li>  
  <li><a href="#">Menu 3</a></li>  
</ul>
```

Display:



Assisted Practice

Forms

Duration: 70 Min.

Problem Statement:

You are given a project to create a form using Bootstrap.

Assisted Practice: Guidelines

Steps to develop a static forms:

1. Build a form using Bootstrap in your code editor
2. View the results in your local browser
3. Push the code to GitHub repositories

Bootstrap: Nav and Navbar

Nav

The **nav** tag is used to define a block of links for navigation, for the current page, or for other pages.

Syntax:

```
<ul class="nav">
  <li class="nav-item">
    <a class="nav-link active" href="#">Active Link1</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Active Link2</a>
  </li>
  <li class="nav-item">
    <a class="nav-link disabled" href="#">Disabled
Link</a>
  </li>
</ul>
```

Display:

Active Link1 Active Link2 Disabled Link

Navigation Bar

A navigation bar is a set of buttons or links used to connect to different sections of the website. Each link can have different features like dropdowns and variation in sizes, colors, and fonts.

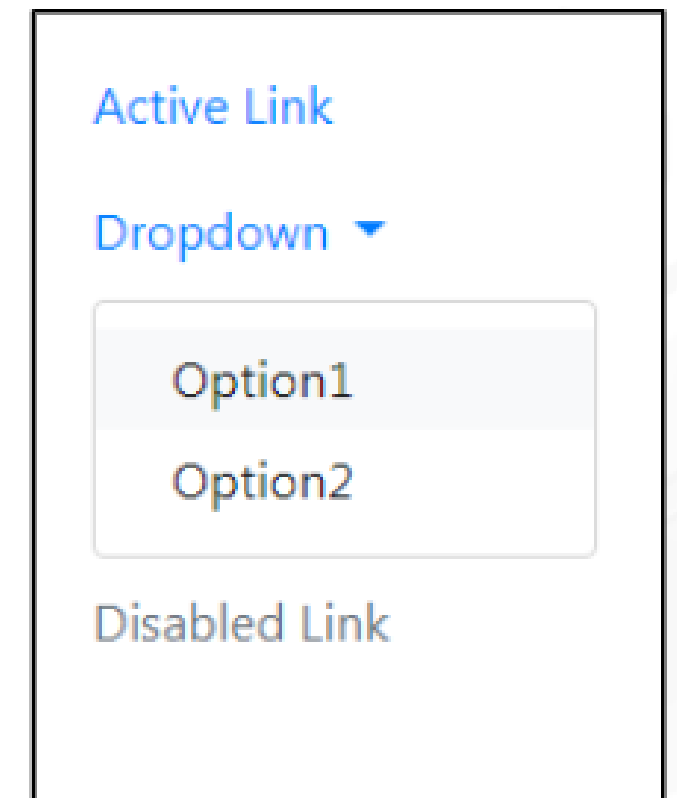
Syntax:

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

    <li class="nav-item dropdown">
      <a class="nav-link dropdown-toggle" href="#" data-toggle="dropdown">
        Dropdown
      </a>
      <div class="dropdown-menu" aria-labelledby="navbarDropdown">
        <a class="dropdown-item" href="#">Option1</a>
        <a class="dropdown-item" href="#">Option2</a>
      </div>
    </li>

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

Display:



Breadcrumbs

A breadcrumb is a scheme for navigation that shows the full path to the current page in the website.

Syntax:

```
<ul class="breadcrumb">
  <li class="breadcrumb-item"><a href="#">Home</a></li>
  <li class="breadcrumb-item"><a href="#">Page1</a></li>
  <li class="breadcrumb-item"><a href="#">Page2</a></li>
  <li class="breadcrumb-item"><a href="#">Page3</a></li>
  <li class="breadcrumb-item active" aria-
current="page">Current Page</li>
</ul>
```

Display:

[Home](#) / [Page1](#) / [Page2](#) / [Page3](#) / Current Page

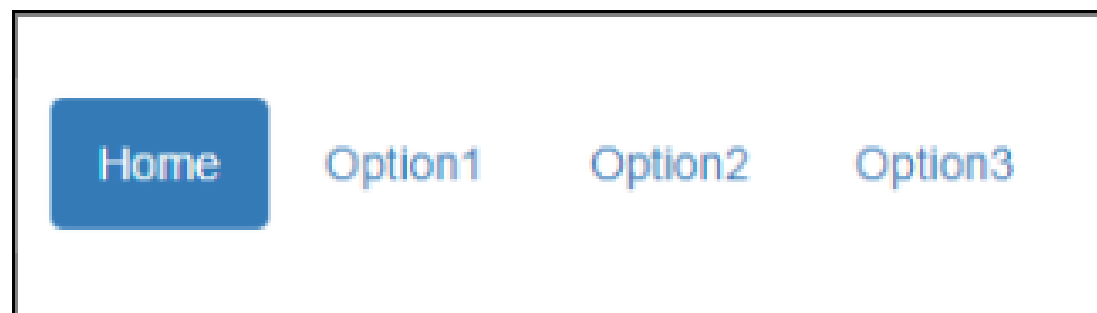
Pills

Pills are like tabs, where the links are placed to navigate to different sections of the website.

Syntax:

```
<ul class="nav nav-pills">  
  <li class="active"><a href="#">Home</a></li>  
  <li><a href="#">Option1</a></li>  
  <li><a href="#">Option2</a></li>  
  <li><a href="#">Option3</a></li>  
</ul>
```

Display:



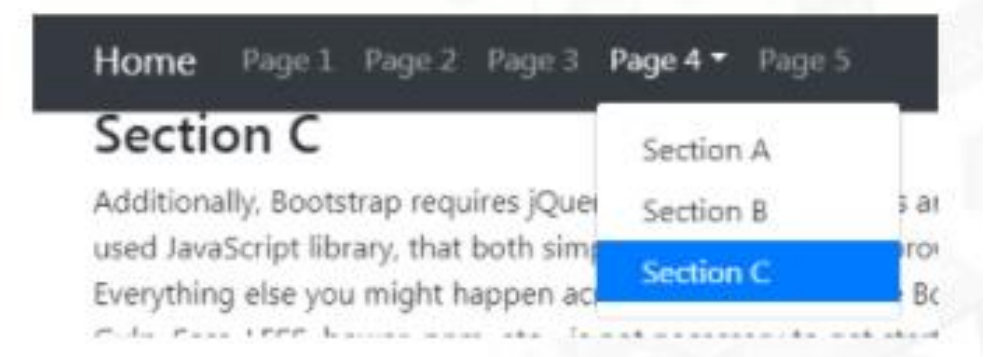
Scrollspy

A scrollspy is a navigation scheme that highlights the navigation links based on the scroll position to show where the user is currently in the page.

Syntax:

```
<body data-spy="scroll">
<a href="#">Home</a>
<ul>
  <li class="nav-item"><a href="#">Page 1</a></li>
  <li class="nav-item"><a href="#">Page 2</a></li>
  <li class="nav-item"><a href="#">Page 3</a></li>
  <li class="nav-item dropdown">
    <a href="#" class="nav-link dropdown-toggle">Section 4</a>
    <div class="dropdown-menu">
      <a href="#" class="dropdown-item">Section A</a>
      <a href="#" class="dropdown-item">Section B</a>
      <a href="#" class="dropdown-item">Section C</a>
    </div>
  </li>
  <li class="nav-item"><a href="#">Page 5</a>
</ul>
```

Display:



Bootstrap: Pagination and Progress

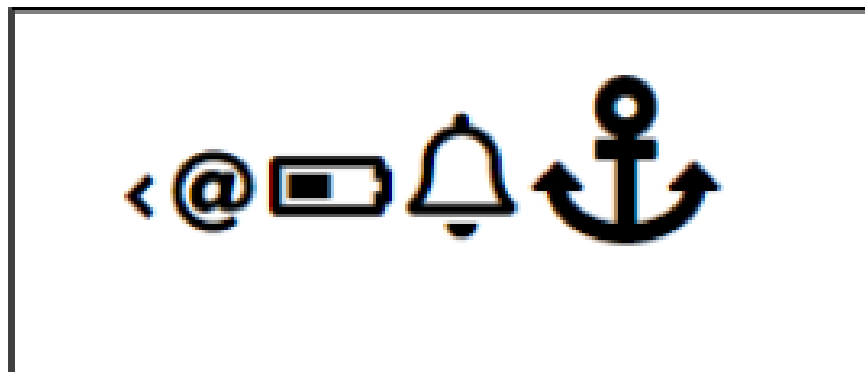
Icons

Icons are images grouped in libraries. Options are provided for better usability and scalability.

Syntax:

```
<i class="fas fa-angle-left" style="font-size:20px"></i>  
<i class="fas fa-at" style="font-size:24px"></i>  
<i class="fas fa-battery-half" style="font-size:28px"></i>  
<i class="far fa-bell" style="font-size:36px"></i>  
<i class="fas fa-anchor" style="font-size:48px"></i>
```

Display:



Pagination

Pagination is used to divide a document into different pages with numbers.

Pagination types	Display
Simple pagination	« 1 2 3 4 »
Active and hoverable pagination	« 1 2 3 4 »
Rounded active and hoverable pagination	« 1 2 3 4 »
Bordered pagination	« 1 2 3 4 »
Rounded border pagination	« 1 2 3 4 »
Space between pagination	« 1 2 3 4 »

Progress Bars

Progress bars are colored stripes that display the amount of progress a user has made in a process.

Syntax:

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

Display:



Spinner

A spinner is a component in HTML and CSS which shows the loading state of the page.

Syntax:

```
<div class="spinner-border text-primary">  
  <span class="sr-only">Loading...</span>  
</div>  
<div class="spinner-border text-secondary">  
  <span class="sr-only">Loading...</span>  
</div>  
<div class="spinner-border text-warning">  
  <span class="sr-only">Loading...</span>  
</div>
```

Display:



Popovers

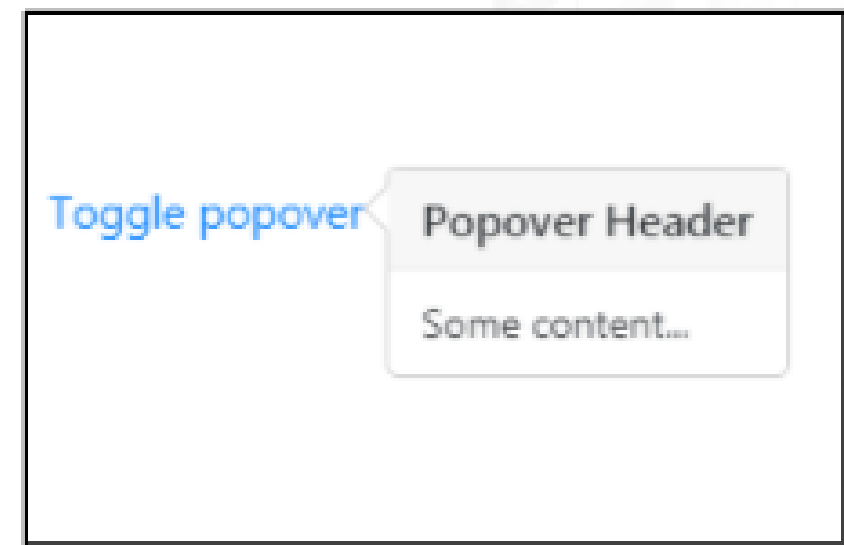
Popovers are used to display additional information about elements when a user hovers over or clicks it. They contain more content than tooltips.

Syntax:

```
<div>
  <a href="#" data-toggle="popover"
  title="Popover Header" data-content="Some
  content...">Toggle popover</a>
</div>

<script>
$(document).ready(function() {
  $('[data-toggle="popover"]').popover();
});
</script>
```

Display:



Assisted Practice

Navigation and Pagination

Duration: 45 Min.

Problem Statement:

You are given a project to add navigation using a navbar and use pagination to navigate in Bootstrap.

Assisted Practice: Guidelines

Steps to implement pagination and navigation to a website:

1. Build a navbar using Bootstrap in your code editor
2. View results in the browser
3. Add pagination to your website using Bootstrap
4. View results in the browser
5. Push the code to GitHub repositories

Key Takeaways

- Responsive Web Design is a responsive browser that can adapt to different screen sizes.
- Bootstrap helps you to create responsive and mobile-friendly websites.
- A navigation bar is a set of buttons or links used to connect to different sections of the website.
- Pagination is used to divide a document into different pages with numbers.
- Tabs are components in which content or links are placed in separate panes to open different pages and sections.



Building a Restaurant Website with Bootstrap

Problem Statement:

You are asked to build a responsive website for a restaurant which provides a brief overview of the restaurant, its bestsellers, its address and the option to contact the restaurant.



Before the next Class

Course:

Full JavaScript Masterclass Course: ES6 Modern Development

You should be able to:

- Setup JavaScript on Windows, Mac OS X, and Linux
- Explain the basic features of JavaScript programming language
- Describe the objects used in JavaScript
- Demonstrate how to create classes in JavaScript

