

## 2. Design Website Using Bootstrap4

---

### What is Bootstrap?

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

### What is Responsive Web Design?

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

Example:

```
<div class="jumbotron text-center">
  <h1>My First Bootstrap Page</h1>
  <p>Resize this responsive page to see the effect!</p>
</div>
```

```
<div class="container">
  <div class="row">
    <div class="col-sm-4">
      <h3>Column 1</h3>
      <p>Loremipsumdolor..</p>
    </div>
    <div class="col-sm-4">
      <h3>Column 2</h3>
      <p>Loremipsumdolor..</p>
    </div>
    <div class="col-sm-4">
      <h3>Column 3</h3>
      <p>Loremipsumdolor..</p>
    </div>
  </div>
</div>
```

## Advantages of Bootstrap:

- Easy to use: Anybody with just basic knowledge of HTML and CSS can start using Bootstrap
- Responsive features: Bootstrap's responsive CSS adjusts to phones, tablets, and desktops
- Mobile-first approach: In Bootstrap 3, mobile-first styles are part of the core framework
- Browser compatibility: Bootstrap is compatible with all modern browsers (Chrome, Firefox, Internet Explorer, Edge, Safari, and Opera)

### jQuery

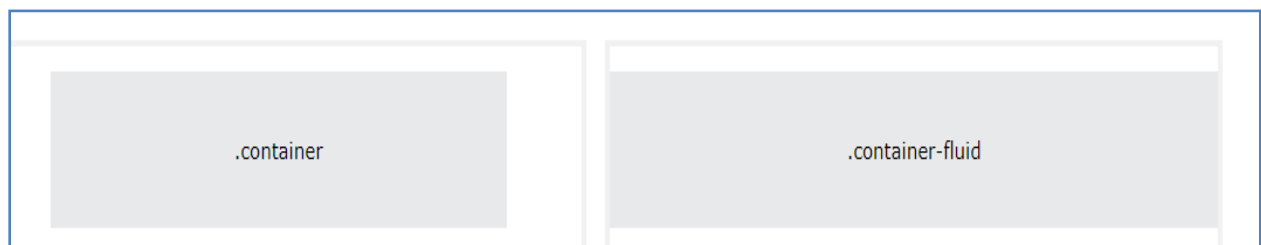
Bootstrap uses jQuery for JavaScript plugins (like modals, tooltips, etc). However, if you just use the CSS part of Bootstrap, you don't need jQuery.

## Containers

Bootstrap 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



## Bootstrap Grid System

Bootstrap's grid system allows up to 12 columns across the page.

If you do not want to use all 12 columns individually, you can group the columns together to create wider columns:

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

Bootstrap's 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:

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

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

## Basic Structure of a Bootstrap Grid

The following is a basic structure of a Bootstrap grid:

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

First, create a row (`<div class="row">`). Then, add the desired number of columns (tags with appropriate `.col-*-*` classes). Note that numbers in `.col-*-*` should always add up to 12 for each row.

### Three Equal Columns

```
.col-sm-4
.col-sm-4
.col-sm-4
```

The following example shows how to get a three equal-width columns starting at tablets and scaling to large desktops. On mobile phones or screens that are less than 768px wide, the columns will automatically stack:

#### Example

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

### Two Unequal Columns

```
.col-sm-4
.col-sm-8
```

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

### Example

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

### Bootstrap Basic Table

A basic Bootstrap table has a light padding and only horizontal dividers.

The `.table` class adds basic styling to a table:

### Example

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

### Striped Rows

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

### Example

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

## Bordered Table

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

### Example

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com

## Contextual Classes

Contextual classes can be used to color table rows (`<tr>`) or table cells (`<td>`):

### Example

Firstname	Lastname	Email
Default	Defaultson	def@somemail.com
Success	Doe	john@example.com
Danger	Moe	mary@example.com
Info	Dooley	july@example.com
Warning	Refs	bo@example.com
Active	Activeson	act@example.com

## Alerts

Bootstrap provides an easy way to create predefined alert messages:

Success! This alert box indicates a successful or positive action.

Info! This alert box indicates a neutral informative change or action.

Warning! This alert box indicates a warning that might need attention.

Danger! This alert box indicates a dangerous or potentially negative action.

Alerts are created with the `.alert` class, followed by one of the four contextual classes `.alert-success`, `.alert-info`, `.alert-warning` or `.alert-danger`:

### Example

```
<div class="alert alert-success">
  <strong>Success!</strong> Indicates a successful or positive action.
</div>
<div class="alert alert-info">
  <strong>Info!</strong> Indicates a neutral informative change or action.
</div>
<div class="alert alert-warning">
  <strong>Warning!</strong> Indicates a warning that might need attention.
</div>
<div class="alert alert-danger">
  <strong>Danger!</strong> Indicates a dangerous or potentially negative
  action.
</div>
```

## Bootstrap Form Layouts

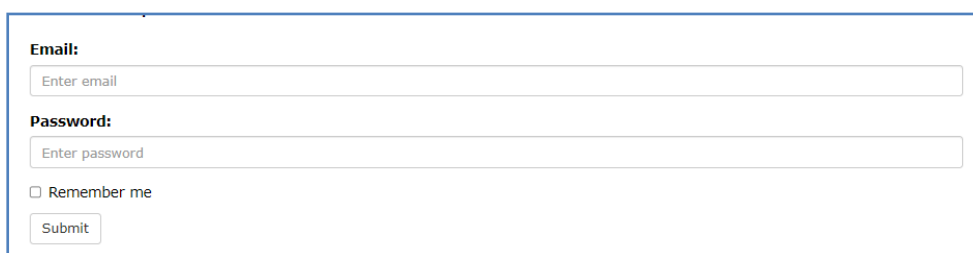
Bootstrap provides three types of form layouts:

- Vertical form (this is default)
- Horizontal form
- Inline form

Standard rules for all three form layouts:

- Wrap labels and form controls in `<div class="form-group">` (needed for optimum spacing)
- Add class `.form-control` to all textual `<input>`, `<textarea>`, and `<select>` elements

### Bootstrap Vertical Form (default)



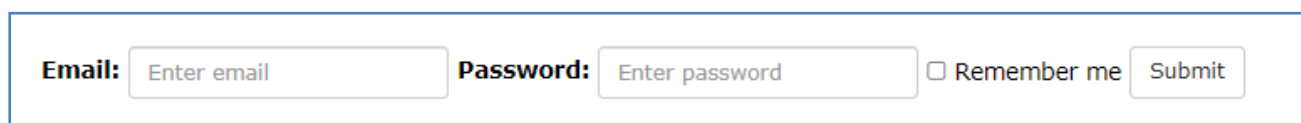
```
<div class="form-group">
  Email:
  <input type="text" value="Enter email" class="form-control"/>
</div>
<div class="form-group">
  Password:
  <input type="password" value="Enter password" class="form-control"/>
  <input type="checkbox" value="Remember me" class="checkbox"/> Remember me
  <input type="submit" value="Submit" class="btn"/>
</div>
```

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

### Example

```
<form action="/action_page.php">
  <div class="form-group">
    <label for="email">Email address:</label>
    <input type="email" class="form-control" id="email">
  </div>
  <div class="form-group">
    <label for="pwd">Password:</label>
    <input type="password" class="form-control" id="pwd">
  </div>
  <div class="checkbox">
    <label><input type="checkbox"> Remember me</label>
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```

### Bootstrap Inline Form



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

Note: This only applies to forms within viewports that are at least 768px wide!

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



```

    <input type="password" class="form-control" id="pwd">
</div>
<div class="checkbox">
    <label><input type="checkbox"> Remember me</label>
</div>
<button type="submit" class="btn btn-default">Submit</button>
</form>

```

## Bootstrap Horizontal Form

A horizontal form means that the labels are aligned next to the input field (horizontal) on large and medium screens. On small screens (767px and below), it will transform to a vertical form (labels are placed on top of each input).

Additional rules for a horizontal form:

- Add class `.form-horizontal` to the `<form>` element
- Add class `.control-label` to all `<label>` elements

Tip: Use Bootstrap's predefined grid classes to align labels and groups of form controls in a horizontal layout.

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

### Example

```

<form class="form-horizontal" action="/action_page.php">
  <div class="form-group">
    <label class="control-label col-sm-2" for="email">Email:</label>
    <div class="col-sm-10">
      <input type="email" class="form-control" id="email" placeholder="Enter
email">
    </div>
  </div>
  <div class="form-group">

```

```

<label class="control-label col-sm-2" for="pwd">Password: </label>
<div class="col-sm-10">
  <input type="password" class="form-control" id="pwd" placeholder="Enter
password">
</div>
</div>
<div class="form-group">
  <div class="col-sm-offset-2 col-sm-10">
    <div class="checkbox">
      <label><input type="checkbox"> Remember me</label>
    </div>
  </div>
</div>
<div class="form-group">
  <div class="col-sm-offset-2 col-sm-10">
    <button type="submit" class="btn btn-default">Submit</button>
  </div>
</div>
</form>

```

## Button Styles

Bootstrap provides different styles of buttons:



To achieve the button styles above, Bootstrap has the following classes:

- `.btn`
- `.btn-default`
- `.btn-primary`
- `.btn-success`
- `.btn-info`
- `.btn-warning`
- `.btn-danger`
- `.btn-link`

The following example shows the code for the different button styles:

### Example

```

<button type="button" class="btn">Basic</button>
<button type="button" class="btn btn-default">Default</button>

```

```
<button type="button" class="btn btn-primary">Primary</button>  
<button type="button" class="btn btn-success">Success</button>  
<button type="button" class="btn btn-info">Info</button>  
<button type="button" class="btn btn-warning">Warning</button>  
<button type="button" class="btn btn-danger">Danger</button>  
<button type="button" class="btn btn-link">Link</button>
```

## Bootstrap Images:

### Rounded Corners

The .img-rounded class adds rounded corners to an image

```

```

### Circle:

The .img-circle class shapes the image to a circle

```

```

### Responsive Images:

Responsive images automatically adjust to fit the size of the screen.

Create responsive images by adding an .img-responsive class to the <img> tag. The image will then scale nicely to the parent element.

The .img-responsive class applies display: block; and max-width: 100%; and height: auto; to the image:

```

```

## Bootstrap Pagination:

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

A basic pagination in Bootstrap looks like this:

To create a basic pagination, add the `.pagination` class to an `<ul>` element:

### Example

```
<ul class="pagination">
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#">4</a></li>
  <li><a href="#">5</a></li>
</ul>
```

### Active State

The active state shows what is the current page:

Add class `.active` to let the user know which page he/she is on:

### Example

```
<ul class="pagination">
  <li><a href="#">1</a></li>
  <li class="active"><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#">4</a></li>
  <li><a href="#">5</a></li>
</ul>
```

### Disabled State

A disabled link cannot be clicked:

Add class `.disabled` if a link for some reason is disabled:

### Example

```
<ul class="pagination">
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li class="disabled"><a href="#">4</a></li>
  <li><a href="#">5</a></li>
</ul>
```

## Pagination Sizing

Pagination blocks can also be sized to a larger size or a smaller size:

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

### Example

```
<ul class="pagination pagination-lg">
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#">4</a></li>
  <li><a href="#">5</a></li>
</ul>
```

```
<ul class="pagination pagination-sm">
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#">4</a></li>
  <li><a href="#">5</a></li>
</ul>
```

## Bootstrap Theme:

### Create a Theme: "Simply Me"

This page will show you how to build a Bootstrap theme from scratch.

We will start with a simple HTML page, and then add more and more components, until we have a fully functional, personal and responsive website.

The result will look like this, and you are free to modify, save, share, use or do whatever you want with it:

### HTML Start Page

We will start with the following HTML page:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Theme Simply Me</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
```

```
</head>
<body>
<h3>Who Am I?</h3>

<h3>I'm an adventurer</h3>
</body>
</html>
```

## Add Bootstrap CDN and Put Elements in Container

Add Bootstrap CDN and a link to jQuery and put HTML elements inside a container:

### Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Theme Simply Me</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/
css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"
></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.mi
n.js"></script>
</head>
<body>

<div class="container-fluid">
  <h3>Who Am I?</h3>
  
  <h3>I'm an adventurer</h3>
</div>

</body>
</html>
```

## Add Background Color and Center Text

1. Add a custom class (.bg-1) to the container to add a background color.
2. Add the .text-center class to center the text inside the container:

### Example

```
<head>
<style>
.bg-1 {
  background-color: #1abc9c; /* Green */
  color: #ffffff;
}
</style>
</head>

<body>
<div class="container-fluid bg-1 text-center">
  <h3>Who Am I?</h3>
  
  <h3>I'm an adventurer</h3>
</div>
</body>
```

## Circle Image

Shape the image to a circle with the .img-circle class:

### Example

```

```

## More Content

Add more content and put it inside new containers:

### Example

```
<head>
<style>
.bg-1 {
  background-color: #1abc9c; /* Green */
  color: #ffffff;
}
.bg-2 {
  background-color: #474e5d; /* Dark Blue */
  color: #ffffff;
}
</style>
</head>
```

```

}
.bg-3 {
  background-color: #ffffff; /* White */
  color: #555555;
}
</style>
</head>

<body>

<div class="container-fluid bg-1 text-center">
  <h3>Who Am I?</h3>
  
  <h3>I'm an adventurer</h3>
</div>
<div class="container-fluid bg-2 text-center">
  <h3>What Am I?</h3>
  <p>Loremipsum..</p>
</div>
<div class="container-fluid bg-3 text-center">
  <h3>Where To Find Me?</h3>
  <p>Loremipsum..</p>
</div>
</body>

```

## Add Padding

Lets make the containers look good by adding some padding:

### Example

```

<style>
.container-fluid {
  padding-top: 70px;
  padding-bottom: 70px;
}
</style>

```



## Add a Button

Add a button to the mid container:

### Example

```
<div class="container-fluid bg-2 text-center">
  <h3>What Am I?</h3>
  <p>Loremipsum..</p>
  <a href="#" class="btn btn-default btn-lg">Search</a>
</div>
```

## Add an Icon

Add a Search icon on the "Search" button:

### Example

```
<a href="#" class="btn btn-default btn-lg">
  <span class="glyphicon glyphicon-search"></span> Search
</a>
```

## Modify the Third Container (Add Grid)

Add three columns of equal width inside the third container (`.col-sm-4`):

### Example

```
<div class="container-fluid bg-3 text-center">
  <h3>Where To Find Me?</h3>
  <div class="row">
    <div class="col-sm-4">
      <p>Loremipsum..</p>
      
    </div>
    <div class="col-sm-4">
      <p>Loremipsum..</p>
      
    </div>
    <div class="col-sm-4">
      <p>Loremipsum..</p>
      
    </div>
  </div>
</div>
```

## Make the Images Responsive

Add the `.img-responsive` class to all images.

Add `display:inline` to the first image to force it to be centered (the `.img-responsive` class adds `display:block` to the image, which makes it jump to the left of the screen).

If you want the image to span the entire width of the screen when it starts to stack, add `width:100%` to the image.

When opening the example, remember to resize the screen to see the responsive effect:

### Example

```
<!-- The circular bird -->


<!-- The birds in our grid: -->



```

## Add a Navbar

Add a standard navigation bar at the top of the page with and make it collapsible on smaller screens:

### Example

```
<nav class="navbar navbar-default">
  <div class="container">
    <div class="navbar-header">
      <button type="button" class="navbar-toggle" data-toggle="collapse" data-
target="#myNavbar">
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="navbar-brand" href="#">Me</a>
    </div>
    <div class="collapse navbar-collapse" id="myNavbar">
      <ul class="nav navbar-nav">
        <li><a href="#">WHO</a></li>
        <li><a href="#">WHAT</a></li>
        <li><a href="#">WHERE</a></li>
      </ul>
    </div>
  </div>
</nav>
```

## Style The Navbar

Use CSS to customize the navigation bar:

### Example

```
.navbar {
  padding-top: 15px;
  padding-bottom: 15px;
  border: 0;
  border-radius: 0;
  margin-bottom: 0;
  font-size: 12px;
  letter-spacing: 5px;
}

.navbar-nav li a:hover {
  color: #1abc9c !important;
}
```

## Add a Footer

Add a footer and use CSS to style it:

### Example

```
<style>
.bg-4 {
  background-color: #2f2f2f;
  color: #ffffff;
}
</style>

<footer class="container-fluid bg-4 text-center">
  <p>Bootstrap Theme Made
  By <a href="https://www.w3schools.com">www.w3schools.com</a> </p>
</footer>
```

## Final Touch

Personalize your Theme by adding a font that you like. We have used "Montserrat" from Google's Font Library.

Link to the font in the `<head>` section:

```
<link href="https://fonts.googleapis.com/css?family=Montserrat" rel="stylesheet">
```

Then you can use it in the page:

### Example

```
body {
  font: 20px "Montserrat", sans-serif;
  line-height: 1.8;
  color: #f5f6f7;
}

p {font-size: 16px;}
```

We have also created a "helper" margin class to add extra space where we think it's needed; usually after each `<h3>` and `<img>` element.

### Example

```
.margin {margin-bottom: 45px;}
```