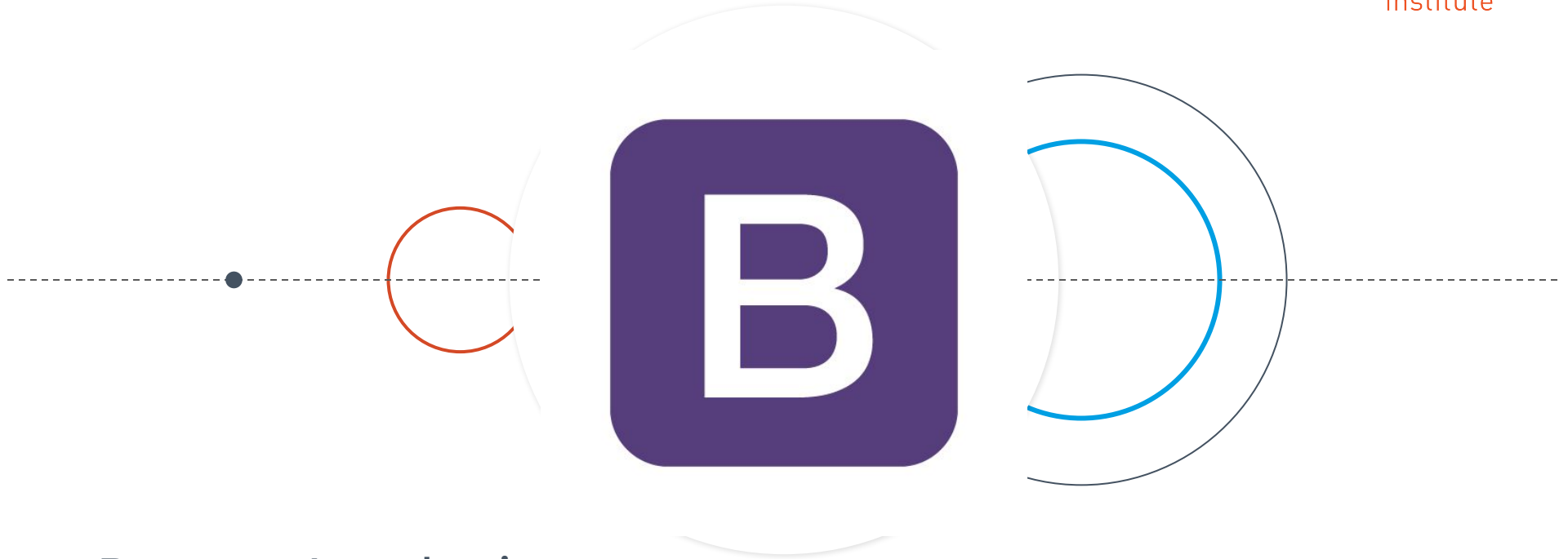
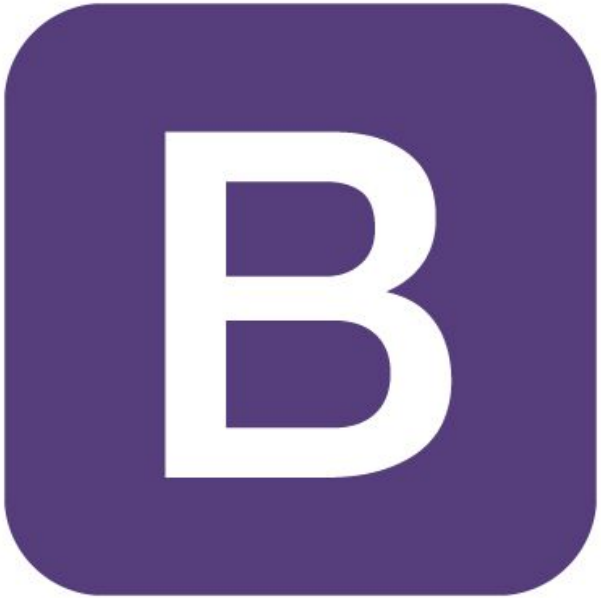
code institute



Bootstrap Introduction

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



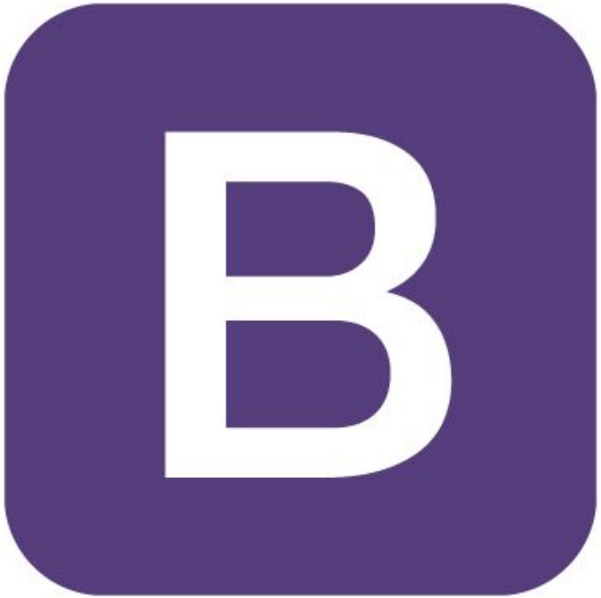
Bootstrap Components

To make your lives easier, bootstrap also has a number of pre-build responsive components, such as:

- Navbars
- Footers
- Accordions
- Buttons
- Carousels

We can use the pre-made HTML found in the bootstrap docs to add into our applications.

So let's take a look at this!



getbootstrap.com

Bootstrap Grid System

By now, you probably already quite a bit about Bootstrap.

But now. remember working on your first project and you had difficulty making your content appear side-by-side on the screen, and stay there no matter what the screen size?

Or maybe, you wanted to stack them on your phone, but have them beside each other on the screen?

Well, that’s where Bootstrap's Grid system comes in incredibly handy!

What is it?

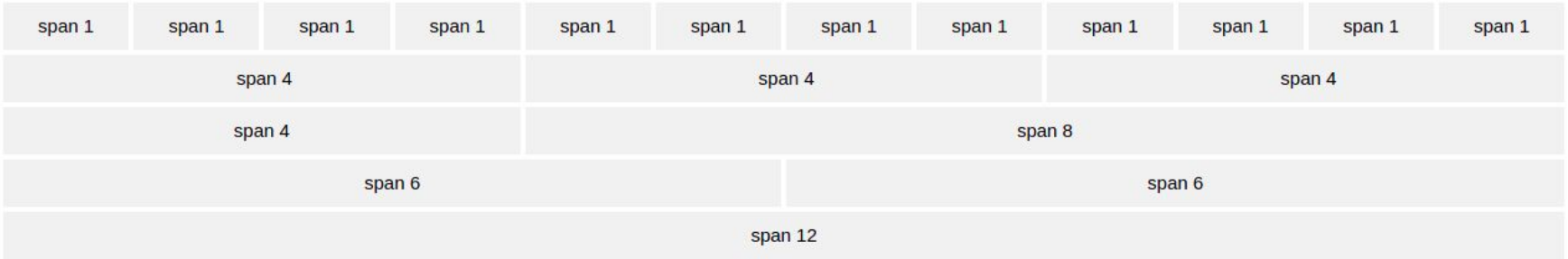
Containers

Rows

Columns

Bootstrap Grid

- Bootstrap's grid system allows up to 12 columns across the page.
- The Bootstrap 4 grid system has five classes:
  - .col- (extra small devices - screen width less than 576px)
  - .col-sm- (small devices - screen width equal to or greater than 576px)
  - .col-md- (medium devices - screen width equal to or greater than 768px)
  - .col-lg- (large devices - screen width equal to or greater than 992px)
  - .col-xl- (xlarge devices - screen width equal to or greater than 1200px)



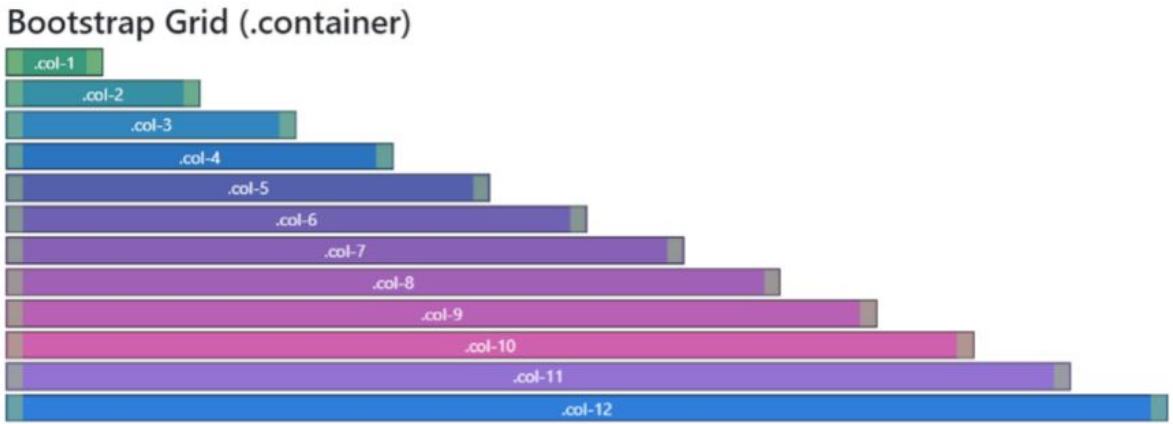
What is the Bootstrap Grid?

We use bootstrap's grid to make responsive layouts.

Bootstraps Grid is made up of a collection of rows and columns within a container.

The grid system is based on 12 column units which are percentage-based, this makes it responsive.

Each column unit is equal to 8.333333% width regardless of device resolutions.



Containers

A **.container** is the most basic element in bootstrap and is required when using the grid system.

Containers are used to contain, pad, and (sometimes) center the content within them. While containers can be nested, most layouts do not require a nested container.

Bootstrap has three different containers:

- **.container**, which sets a max-width at each responsive breakpoint
- **.container-fluid**, which is width: 100% at all breakpoints
- **.container-{breakpoint}**, which is width: 100% until the specified breakpoint

	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	X-Large ≥1200px	XX-Large ≥1400px
`.container`	100%	540px	720px	960px	1140px	1320px
`.container-sm`	100%	540px	720px	960px	1140px	1320px
`.container-md`	100%	100%	720px	960px	1140px	1320px
`.container-lg`	100%	100%	100%	960px	1140px	1320px
`.container-xl`	100%	100%	100%	100%	1140px	1320px
`.container-xxl`	100%	100%	100%	100%	100%	1320px
`.container-fluid`	100%	100%	100%	100%	100%	100%

Rows

Rows are wrappers for columns.

Each column has horizontal padding (called a gutter) for controlling the space between them. This padding is then counteracted on the rows with negative margins to ensure the content in your columns is visually aligned down the left side.

Rows also support modifier classes to uniformly apply column sizing and gutter classes to change the spacing of your content.

```
<div class="container">
  <div class="row">
    <div class="col">
      Column
    </div>
    <div class="col">
      Column
    </div>
    <div class="col">
      Column
    </div>
  </div>
</div>
```

Columns

The hierarchy of Bootstrap's grid goes from container to row to column to your content.

There are 12 template columns available per row (1 - 12), allowing you to create different combinations of elements that span any number of columns.

Column classes indicate the number of template columns to span and widths are set in percentages so you always have the same relative sizing.

There are 6 default grid tiers to each template column, which means that you can customise your application responsiveness to work across 6 different screen sizes. We do generally use three of these 6, **sm, md, lg.**

	xs <576px	sm ≥576px	md ≥768px	lg ≥992px	xl ≥1200px	xxl ≥1400px
Container `max-width`	None (auto)	540px	720px	960px	1140px	1320px
Class prefix	`.col-`	`.col-sm-`	`.col-md-`	`.col-lg-`	`.col-xl-`	`.col-xxl-`
# of columns	12					
Gutter width	1.5rem (.75rem on left and right)					
Custom gutters	Yes					
Nestable	Yes					
Column ordering	Yes					

Other Tricks with Bootstrap Grid

On top of creating a system based on the grid, we can also offset columns. This means based on the screen sizes we can specify exactly where we want our column to lay within the 12 template sizes.

Move columns to the right using .offset-md-* classes. These classes increase the left margin of a column by * columns.

For example, .offset-md-4 moves .col-md-4 over four columns, we can see this in the first row.

Copy

```
<div class="container">
  <div class="row">
    <div class="col-md-4">.col-md-4</div>
    <div class="col-md-4 offset-md-4">.col-md-4 .offset-md-4</div>
  </div>
  <div class="row">
    <div class="col-md-3 offset-md-3">.col-md-3 .offset-md-3</div>
    <div class="col-md-3 offset-md-3">.col-md-3 .offset-md-3</div>
  </div>
  <div class="row">
    <div class="col-md-6 offset-md-3">.col-md-6 .offset-md-3</div>
  </div>
</div>
```