



Bootstrap 4 Framework CSS Grid System

Paul Bennett

CODE INSTITUTE (FIRST CLASS HONOURS ALUMNI)

## Table of Contents

What is Bootstrap?.....	1
What is Bootstrap Grid?.....	2
What are Bootstrap Grid Breakpoints?.....	2
How does .row work in Bootstrap 4?.....	3
How does .col work in Bootstrap4 ?.....	4

## What is Bootstrap?

Bootstrap 4 is a CSS(Cascading Style Sheets) framework built with flexbox and used for building responsive, mobile-first websites.

A CSS software framework is used to help you style your website, and provide a rich set of pre-styled classes and CSS classes which you can add to your HTML.

Bootstrap covers mainly three areas:

- General Default Styles
  - **Reboot** (Sets cross-browser style defaults)
  - **Typography/Default Styles** (Font family, sizes & weights)
  - **Tables, Images, Figures** (Default responsive styles)
- Pre-defined CSS Classes / Components
  - Alerts, Buttons, Card, Forms, Navbar etc
- Grid System Layout
  - Container > Row(s) > Column(s)

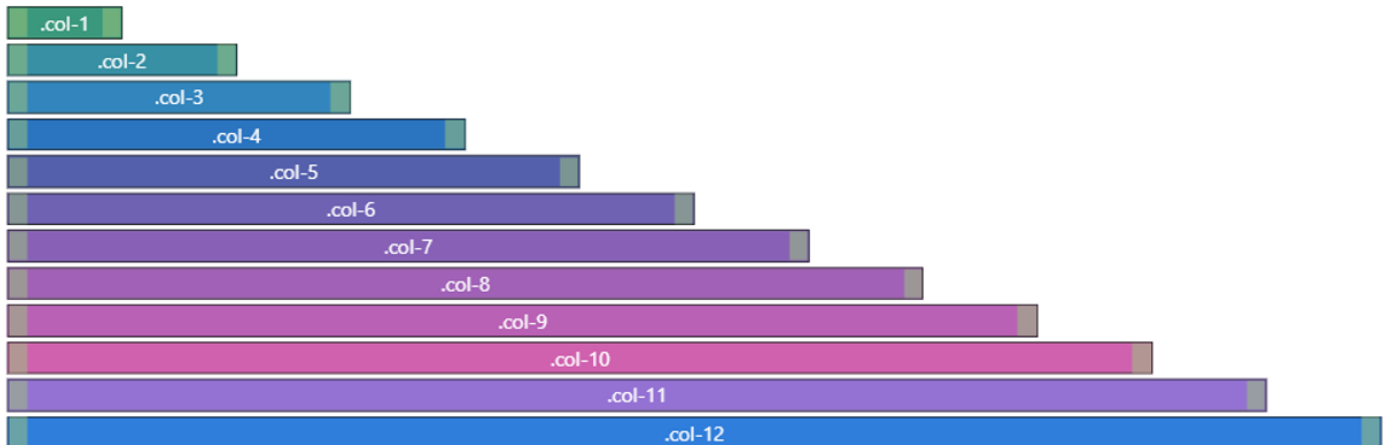
A responsive web design represents a way to make your website look good on multiple device resolutions i.e. Phones, Tablets, Desktops.

## What is Bootstrap Grid?

The Bootstrap grid system is used for responsive layouts, made up of combinations of `.rows` & `.columns` inside one or more `.container` classes.

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.

### Bootstrap Grid (`.container`)



Grid columns are created by specifying the # of 12 available column units you wish to use.

## What are Bootstrap Grid Breakpoints?

The grid system has five responsive breakpoints:

- `.col-` extra small devices (screen width lt 576px)
- `.col-sm-` small devices (screen width eq or gt 576px)
- `.col-md-` medium devices (screen width eq or gt 768px)
- `.col-lg-` large devices (screen width eq or gt 992px)
- `.col-xl-` xlarge devices (screen width eq or gt 1200px)

Columns having the classes `.col-1` to `.col-12` are the “default” classes.

Each grid class, scales UP, So if you wish to set the same widths for small & medium devices, you only need to specify `col-sm-`

A `.container` is the most basic element in bootstrap4 and is required when using the grid system.

On viewing the `bootstrap.css` file the first declaration (line 548-554) is one for `.container` class, this means that the container with this class will have a width of 100%, left & right padding of 15px and auto left and right margins on all devices.

```

548 .container {
549     width: 100%;
550     padding-right: 15px;
551     padding-left: 15px;
552     margin-right: auto;
553     margin-left: auto;
554 }
555
556 @media (min-width: 576px) {
557     .container {
558         max-width: 540px;
559     }
560 }
561
562 @media (min-width: 768px) {
563     .container {
564         max-width: 720px;
565     }
566 }
567
568 @media (min-width: 992px) {
569     .container {
570         max-width: 960px;
571     }
572 }
573
574 @media (min-width: 1200px) {
575     .container {
576         max-width: 1140px;
577     }
578 }

```

### *CSS Code from Bootstrap.css*

The first `.container` class declaration (line 548-554) was made outside the first media query `@media (min-width: 576px)` (line 556) which means all devices with a width smaller than 576px, so is the default and that's why bootstrap is known as mobile-first.

- `.col-` container width none (auto)
- `.col-sm-` container width (max 540px)
- `.col-md-` container width (max 720px)
- `.col-lg-` container width (max 960px)
- `.col-xl-` container width (max 1140px)

All above have Gutter width of 30px (15px on each side of column).

### How does `.row` work in Bootstrap 4?

Bootstrap 4 uses flexbox which is a new layout mode of CSS3 and is officially called CSS Flexible Box Layout Module.

It defines a CSS box model optimized for user interface design, and the layout in terms of items in one dimension.

See [https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Flexible\\_Box\\_Layout](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout) for more info.

```

587
588 .row {
589     display: -ms-flexbox;
590     display: flex;
591     -ms-flex-wrap: wrap;
592     flex-wrap: wrap;
593     margin-right: -15px;
594     margin-left: -15px;
595 }
596

```

*CSS Code from Bootstrap.css*

`.row` class uses `flex-wrap: wrap;` (line 592) which means that the items inside it will wrap so they will go one below the other when the browser window is resized.

`margin-right: -15px;` & `margin-left: -15px;` (lines 593 & 594) these counteract the padding in the `.container` class (lines 550 & 551) and all the content in the columns is visually aligned down the left-hand side.

### How does `.col` work in Bootstrap4 ?

```

619
620 .col {
621     -ms-flex-preferred-size: 0;
622     flex-basis: 0;
623     -ms-flex-positive: 1;
624     flex-grow: 1;
625     max-width: 100%;
626 }
627

```

*CSS Code from Bootstrap.css*

If we have two columns with `.col` classes, they will adjust, and each column will be **50%** of the `.row` width.

If we have four columns with `.col` classes, they will adjust, and each column will be **25%** of the `.row` width.

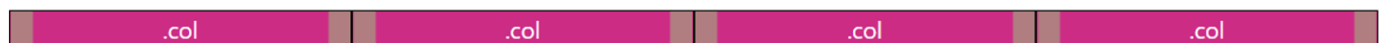
This is because they are flex-based and because of `flex-grow: 1;` (line 624) and `flex-basis: 0;` (line 622) all `.col` columns in a `.row` will always have equal widths.

This class does not have any breakpoints, so will behave the same on all devices.

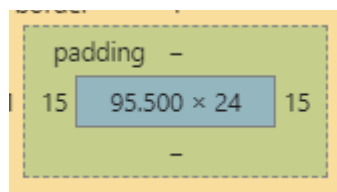
## Here's the example with 4 columns

```
56
57     <div class="container">
58         <h2>Basic Columns (.container)</h2>
59         <div class="row">
60             <div class="col">.col</div>
61             <div class="col">.col</div>
62             <div class="col">.col</div>
63             <div class="col">.col</div>
64         </div>
65     </div>
66
```

## Basic Columns (.container)



The boxes at the ends of each column are the 15px padding.



And as you can see has `padding-right: 15px;` & `padding-left: 15px;` for each column as per `.container` class (lines 550 & 551).

## Here's an example with 3 columns

`.col-1` to `.col-12` all perform the same on all devices and each one has a certain width defined in `bootstrap.css` file as listed below.

Column					Width %
.col-1	.col-sm-1	.col-md-1	.col-lg-1	.col-xl-1	<code>max-width: 8.333333%;</code>
.col-2	.col-sm-2	.col-md-2	.col-lg-2	.col-xl-2	<code>max-width: 16.666667%;</code>
.col-3	.col-sm-3	.col-md-3	.col-lg-3	.col-xl-3	<code>max-width: 25%;</code>
.col-4	.col-sm-4	.col-md-4	.col-lg-4	.col-xl-4	<code>max-width: 33.333333%;</code>
.col-5	.col-sm-5	.col-md-5	.col-lg-5	.col-xl-5	<code>max-width: 41.666667%;</code>
.col-6	.col-sm-6	.col-md-6	.col-lg-6	.col-xl-6	<code>max-width: 50%;</code>
.col-7	.col-sm-7	.col-md-7	.col-lg-7	.col-xl-7	<code>max-width: 58.333333%;</code>
.col-8	.col-sm-8	.col-md-8	.col-lg-8	.col-xl-8	<code>max-width: 66.666667%;</code>
.col-9	.col-sm-9	.col-md-9	.col-lg-9	.col-xl-9	<code>max-width: 75%;</code>
.col-10	.col-sm-10	.col-md-10	.col-lg-10	.col-xl-10	<code>max-width: 83.333333%;</code>
.col-11	.col-sm-11	.col-md-11	.col-lg-11	.col-xl-11	<code>max-width: 91.666667%;</code>
.col-12	.col-sm-12	.col-md-12	.col-lg-12	.col-xl-12	<code>max-width: 100%;</code>

All these widths are the same for other classes (sm, md, lg, xl), but perform differently as they are defined in the `@media` queries (lines 556 – 578) in the `bootstrap.css` file

```
56
57     <div class="container">
58         <h2>Set Sizes (.container)</h2>
59         <div class="row">
60             <div class="col-5">.col-5</div>
61             <div class="col-3">.col-3</div>
62             <div class="col-4">.col-4</div>
63         </div>
64     </div>
65
```

## Set Sizes (.container)



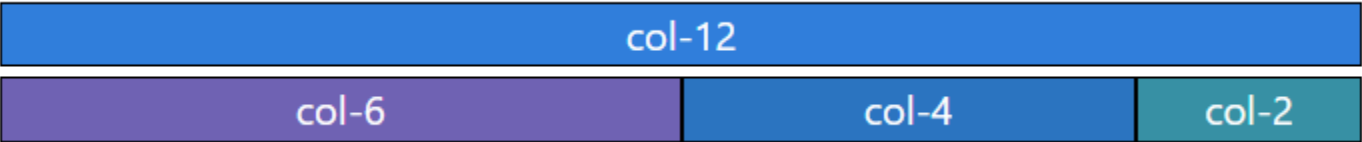
The boxes at the ends of each column are the 15px padding.

Here’s an example with a combination of different classes/columns for each breakpoint

```
86     <div class="container">
87         <h2>Combo Columns (.container)</h2>
88         <div class="row">
89             <div class="col-12 col-sm-12 col-cols">col-sm-12</div>
90             <div class="col-6 col-sm-2 col-md-6 col-lg-2 col-xl-6 col-cols">col-xl-6</div>
91             <div class="col-4 col-sm-6 col-md-4 col-lg-4 col-xl-4 col-cols">col-xl-4</div>
92             <div class="col-2 col-sm-4 col-md-2 col-lg-6 col-xl-2 col-cols">col-xl-2</div>
93         </div>
94     </div>
```

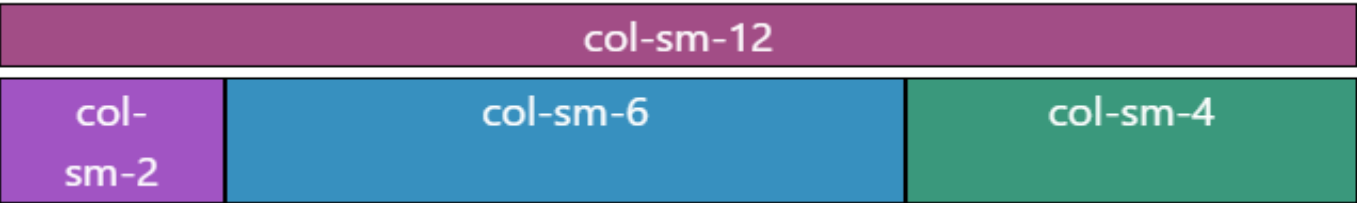
Colours of columns change with each breakpoint

(Extra Small Devices) less than 576px



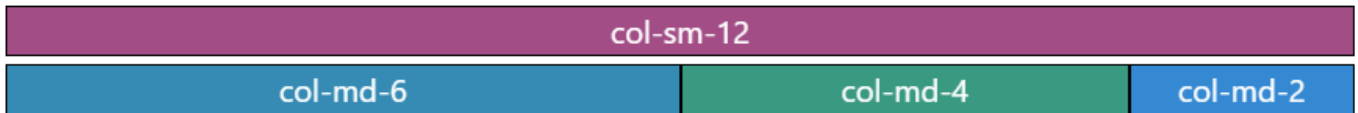
(Small Devices) equal or greater than 576px

@media screen and (min-width:576px)



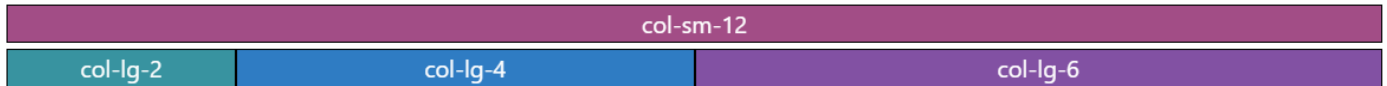
## (Medium Devices) equal or greater than 768px

@media screen and (min-width: 768px)



## (Large Devices) equal or greater than 992px

@media screen and (min-width: 992px)



## (Extra Large Devices) equal or greater than 1200px

@media screen and (min-width: 1200px)



If we inspect the chrome developer you see as per (line 92) in the code the grid calculations for each breakpoint.

```
@media (min-width: 1200px)
.col-xl-2 {
  -ms-flex: 0 0 16.666667%;
  flex: 0 0 16.666667%;
  max-width: 16.666667%;
}

@media (min-width: 992px)
.col-lg-6 {
  -ms-flex: 0 0 50%;
  flex: 0 0 50%;
  max-width: 50%;
}

@media (min-width: 768px)
.col-md-2 {
  -ms-flex: 0 0 16.666667%;
  flex: 0 0 16.666667%;
  max-width: 16.666667%;
}

@media (min-width: 576px)
.col-sm-4 {
  -ms-flex: 0 0 33.333333%;
  flex: 0 0 33.333333%;
  max-width: 33.333333%;
}

.col-2 {
  -ms-flex: 0 0 16.666667%;
  flex: 0 0 16.666667%;
  max-width: 16.666667%;
}
```

## Credits

## Reference Material

<https://www.zeninvader.com/css/understanding-the-bootstrap-4-framework-css-grid-system>