



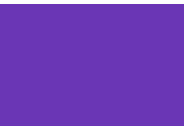
Bootstrap Class Day 2

#Bootstrap Notes

Responsive breakpoints

- Since Bootstrap is developed to be mobile first, we use a handful of media queries to create sensible breakpoints for our layouts and interfaces. These breakpoints are mostly based on minimum viewport widths and allow us to scale up elements as the viewport changes.
- Bootstrap primarily uses the following media query ranges—or breakpoints—in our source Sass files for our layout, grid system, and components.





```
// Extra small devices (portrait phones, less than 576px)
// No media query since this is the default in Bootstrap
```

```
// Small devices (landscape phones, 576px and up)
@media (min-width: 576px) { ... }
```

```
// Medium devices (tablets, 768px and up)
@media (min-width: 768px) { ... }
```

```
// Large devices (desktops, 992px and up)
@media (min-width: 992px) { ... }
```

```
// Extra large devices (large desktops, 1200px and up)
@media (min-width: 1200px) { ... }
```



Bootstrap Grid Introduction



Bootstrap Grid Introduction

- **Bootstrap 4** provides a useful tool for creating responsive layouts - the **grid system**.
- The grid system of Bootstrap 4 is **built with flexbox** and allows **up to 12** columns for laying out your website content.
- The **main advantage** of the Bootstrap 4 grid system is that it will **rearrangedepending on the screen size**, making it perfect for responsive web design.



col 1	col 1	col 1	col 1	col 1	col 1	col 1	col 1	col 1	col 1	col 1	col 1
col 3			col 3			col 3			col 3		
col 4				col 4				col 4			
col 8								col 4			
col 2		col 2		col 4				col 2		col 2	
col 6						col 6					
col 12											



Bootstrap Grid Classes

- Bootstrap 4 has included 5 predefined grid classes to scale the content depending on the device or viewport size.
 - .col-*
 - .col-sm-*
 - .col-md-*
 - .col-lg-*
 - .col-xl-*
-
- Here, the asterisk (*) is the span width of the column from **1** to **12**. The following table lists how the grid system classes will work across multiple devices.



Class	Device Type	Width
.col-*	Extra Small	<576px
.col-sm-*	Small	≥576px
.col-md-*	Medium	≥768px
.col-lg-*	Large	≥992px
.col-xl-*	Extra Large	≥1200px



Bootstrap 4 Grid Introduction Classes

- There are five classes in the Bootstrap 4 stylesheet, that are associated with the grid system.
- **.col-** is used for extremely small devices, which have the screen width of less than 576px.
- **.col-sm-** is used for small devices, which have the screen width of 576px or more.
- **.col-md-** is used for medium sized devices, which have the screen width of 768px or more.
- **.col-lg-** is used for large devices, which have the screen width of 992px or more.
- **.col-xl-** is used for extra large small devices, which have the screen width of 1200px or more.



Structure of Bootstrap Grid

- To create responsive web page layouts using a bootstrap grid system, we need to use rows and columns within the container or container-fluid like as shown below.

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



- We created a grid layout by specifying the rows and columns within `.container`. Here, the first star (*) will indicate the responsiveness, and that should be either `sm`, `md`, `lg`, or `xl`. The second star (*) will indicate a number, and that should be from 1 to 12 for each row.
- Every time while we create a grid layout, we need to place the content within the columns (`.col` or `.col-*-*`) and those columns must be the children of rows (`.row`) and those rows must be placed inside of a container (`.container` or `.container-fluid`).
- In the bootstrap grid system, we are allowed to add up to 12 columns within a row. If we add more than 12 columns within a row, those extra columns will wrap into a new line.



Bootstrap Grid Layout with Responsive Three Column

- Following is the example of creating a responsive layout with three columns to support the devices whose screen width is more than 576px in bootstrap.

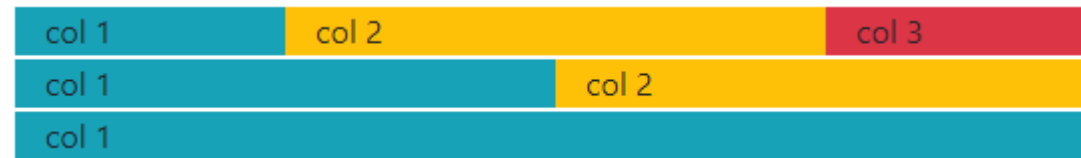
```
<div class="container">
  <div class="row">
    <div class="col-sm-4">col-sm-4</div>
    <div class="col-sm-4">col-sm-4</div>
    <div class="col-sm-4">col-sm-4</div>
  </div>
  <div class="row">
    <div class="col-sm-2">col-sm-2</div>
    <div class="col-sm-4">col-sm-4</div>
    <div class="col-sm-6">col-sm-6</div>
  </div>
  <div class="row">
    <div class="col-sm-2">col-sm-2</div>
    <div class="col-sm-5">col-sm-5</div>
    <div class="col-sm-5">col-sm-5</div>
  </div>
</div>
```



Bootstrap Grid Layout with Auto Columns

- In bootstrap, we can create equal or auto width columns for all the devices such as extra small, small, medium, large, and extra-large by specifying only .col class without any column numbers.
- The bootstrap grid system will automatically adjust the columns defined with .col class based on the device screen width.

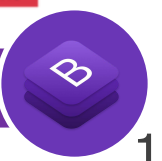
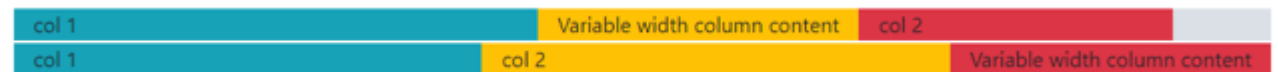
```
<div class="container">
  <div class="row">
    <div class="col">col 1</div>
    <div class="col">col 2</div>
    <div class="col">col 3</div>
  </div>
  <div class="row">
    <div class="col">col 1</div>
    <div class="col">col 2</div>
  </div>
  <div class="row">
    <div class="col">col 1</div>
  </div>
</div>
```



Bootstrap Grid Layout with Variable Width Columns

- In bootstrap, by using col-{breakpoint}-auto classes, we can create variable-width columns to resize based on the width of their content.
- Following is the example of creating variable width columns to resize based on the width of content in bootstrap.

```
<div class="container">
  <div class="row">
    <div class="col-sm-5">col 1</div>
    <div class="col-md-auto">Variable width column content</div>
    <div class="col-sm-3">col 2</div>
  </div>
  <div class="row">
    <div class="col">col 1</div>
    <div class="col">col 2</div>
    <div class="col-md-auto">Variable width column content</div>
  </div>
</div>
```



Bootstrap Grid Layout Column Offsetting

- In the bootstrap grid layout, we can move columns to the right by using `.offset-*` grid classes. To offset a column, you have to append `offset-*` suffix to your existing column class.
- For example, if your column class is, `.col-md-4` you can offset the column with the class `.offset-md-4`. The number after offset symbolizes how many columns to add as margin.

```
<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>
```

.col-md-4

.col-md-3 .offset-md-3

.col-md-6 .offset-md-3

.col-md-4 .offset-md-4

.col-md-3 .offset-md-3



Bootstrap Grid Layout Column Reordering

- we can reorder the appearance of grid columns by using `.order-*` classes without changing their actual order in a grid layout.
- In the grid layout, the order of grid columns will vary based on the defined order numbers. For example, the grid columns with lower-order (e.g., `.col`, `.order-1`) or without order classes will appear first, and the columns with higher-order numbers will appear after the lower order number columns. The grid system will support ordering the columns from 1 through 12 across all five grid tiers.

```
<div class="container">  
  <div class="row">  
    <div class="col order-10">First</div>  
    <div class="col order-4">Second</div>  
    <div class="col">Third</div>  
  </div>  
</div>
```



- <https://getbootstrap.com/docs/4.0/layout/grid/>

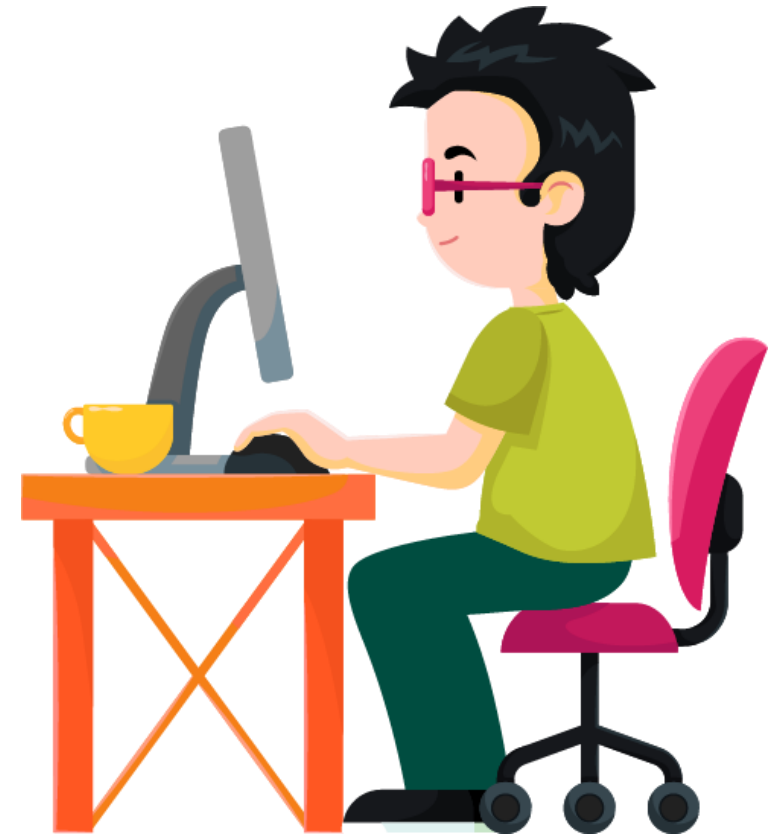


Get Exclusive Video Tutorials



www.apptutorials.com

<https://www.youtube.com/user/Akashtips>





Get More Details

www.akashsir.com

If You Liked It !

Rating Us Now



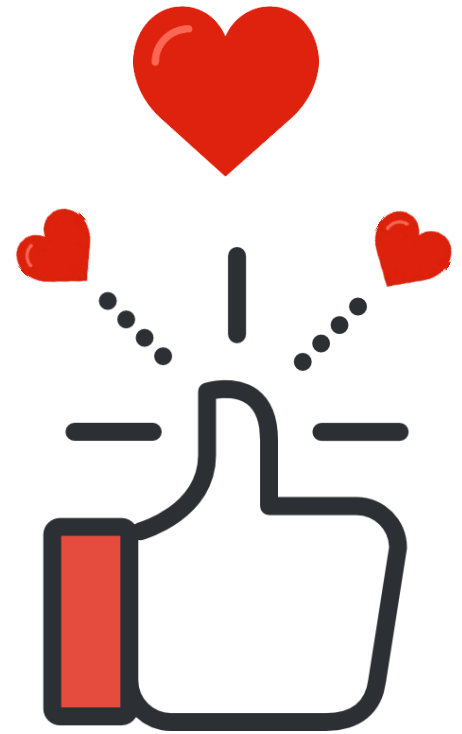
Just Dial

https://www.justdial.com/Ahmedabad/Akash-Technolabs-Navrangpura-Bus-Stop-Navrangpura/079PXX79-XX79-170615221520-S5C4_BZDET



Sulekha

<https://www.sulekha.com/akash-technolabs-navrangpura-ahmedabad-contact-address/ahmedabad>



Connect With Me



Akash Padhiyar
#AkashSir

www.akashsir.com
www.akashtechlabs.com
www.akashpadhiyar.com
www.apptutorials.com

Social Info



Akash.padhiyar



Akashpadhiyar



Akash_padhiyar



+91 99786-21654



#Akashpadhiyar
#apptutorials