Bootstrap grid:

**2 column layout**

<div class="**container**">   
<div class="**row**">   
<div class="**col**">Left column</div>  
<div class="**col**">Right column</div>   
</div>   
</div>

**3 column layout**

<div class="**container**">   
<div class="**row**">   
<div class="**col**">Left column</div>  
<div class="**col**">Center column</div>  
<div class="**col**">Right column</div>   
</div>   
</div>

RULES:

Columns **must** be the immediate child of a Row.

Rows are **only** used to contain Columns, nothing else

rows should be placed inside a Container.

**<div class="container">**  
**<h2>My Heading</h2>**   
**<div class="row">**   
<div class="col">I'm content inside the grid!</div>   
**</div>**   
**</div>**

**By Default .row has {display:flex}**

**As per flex,** Columns in each row are the **same height**

**Columns :**

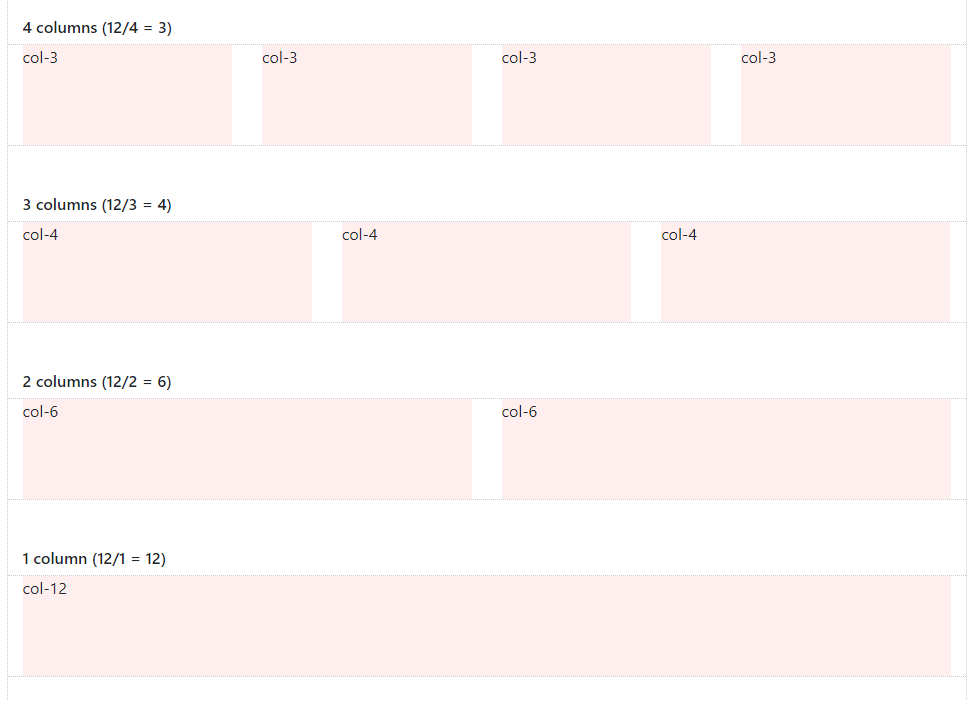
* Create horizontal divisions across the viewport.
* Can have different defined widths.
* Can change in width at different screen widths.
* Layout *horizontally* left-to-right, then *vertically* up-and-down.
* Can change position (re-order) relative to siblings in the same row.
* Are the same height as other siblings in the same row.
* Can “grow” or “shrink” in width.
* Can automatically “wrap” or “stack” vertically as needed, or at different screen widths.
* Can contain *more* Rows & Columns\_\_ nesting.

Columns create **horizontal divisions** across the viewport.

The space between the columns is called the “**gutter**”**.**

The columns can be split-up using any part of the 12 units. And, **it’s okay to use less than 12.**It’s also okay to use **more than 12** which I’ll show you later.

We can split it into



We can have ‘n’ number of combinations

The Grid is not always about 12.

We have [Flexbox](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout) - **“[Auto-layout](http://getbootstrap.com/docs/4.1/layout/grid/" \l "auto-layout-columns" \t "_blank)” Columns**

The Auto-layout Columns have responsive tiers (-sm,-md,-lg,-xl) .

 Columns can be different widths:

**Column width** can **change depending on screen**

**Vertical “stacking” or “wrapping”** occurs when the Column units in a single Row **exceed 12 – “column wrapping”**

**BootStrap 4 has 5 responsive tiers:**

**Responsive Breakpoints, based on screen width:**

* (xs) — screen width < **576**px (This is the “default” tier)(break pt)
* sm — screen width ≥ **576**px
* md — screen width ≥**768**px
* lg — screen width ≥**992**px
* xl — screen width ≥ **1200**px

Bootstrap uses media queries for responsive design and that’s why we are able to see the components adjusting or wrapping

Ex:

<div class="**container**">   
<div class="**row**">   
<div class="**col-sm-6**">Column 1</div>  
<div class="**col-sm-6**">Column 2</div>   
</div>   
</div>

Here col-sm-6 is used twice and use 50% for each div

And it wraps or breaks if size is lessthan 768

## Larger breakpoints, override Smaller breakpoints.

xs(default) >  
overridden by sm >  
overridden by md >  
overridden by lg >  
overridden by xl

**Or,** in reverse…  
xl > overrides lg > overrides md > overrides sm > overrides (xs)

<div class="**col-lg-3 col-md-3 col-sm-3**">..</div> is the same as,

<div class="**col-sm-3**">..</div>

**Auto-layout Grid :**

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

Combination of traditional + auto-layout can also be done:

<div class="container">

<div class="row">

<div class="col-sm-2" style="background-color:yellow;">sidebar</div>

<div class="col" style="background-color:orange;">main content</div>

</div>

</div>

So here col-sm-2 has a specified size in and other column has “col “ which Is responsive[flex] fills the dev

If screen if goes less than the device size

If size is increased then col1 will have less space and col2 will have more space

<div class="**container**">   
<div class="**row**">   
<div class="**col-sm-2**">left</div>  
<div class="**col**">main content</div>  
<div class="**col-auto**">right</div>  
</div>   
</div>