



What's New in the Bootstrap 4 Grid

November 15th 2016 · **Bootstrap** (<https://tutorialzine.com/tag/bootstrap>) | **Flexbox** (<https://tutorialzine.com/tag/flexbox>)

Demo (<https://demo.tutorialzine.com/2016/11/boostrap-4-regular-vs-flex-grid/>)

Bootstrap 4 brings many changes and new features to the grid system we are all so familiar with from version 3. The new grid is now powered by flexbox, lots of utility classes have been renamed, and a new XL breakpoint has been added.

When we first wrote this article, Bootstrap 4 was supposed to have two separate grids. Shortly after we published it, a new Alpha (<https://blog.getbootstrap.com/2017/01/06/bootstrap-4-alpha-6/>) was released, making flexbox the only layout mode of the framework as well as introducing some new features. We have updated the article to reflect those changes.

1. Basic Grid

By now everyone knows how the Bootstrap grid works. We've got rows separated into 12 equal pieces, and columns that go inside the rows. Each column can take anywhere from 1 to 12

spaces:



```

<div class="row">
  <div class="col-xs-2">.col-xs-2</div>
  <div class="col-xs-4">.col-xs-4</div>
  <div class="col-xs-6">.col-xs-6</div>
</div>

```

Structurally nothing has changed, the grid still has rows and 12 columns. However, there are changes in the width of containers, as well as other small stuff like the lowest breakpoint tier being renamed from `.col-xs-` to simply `.col-`

```

<div class="row">
  <div class="col-2">.col-2</div>
  <div class="col-4">.col-4</div>
  <div class="col-6">.col-6</div>
</div>

```

To help you better visualize the changes, we've prepared side-by-side demos (<https://demo.tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/?1-basic-grid>) of both new flex and old non-flex grids. You can check them out below:



(<https://demo.tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/?1-basic-grid>)



(<https://www.facebook.com/sharer/sharer.php?u=https://tutorialzine.com/2016/11/bootstrap-4-regular->

A Basic Bootstrap Grid

2. Automatic Layout

A cool new feature of the Bootstrap 4 grid is the auto-layout mode. It lets developers leave out the size of columns, making them automatically distribute the space in that row.

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

Sizeless columns share the available space equally, always filling up the entire row. If we want a column to be bigger or smaller, we can still do that with a `.col-size` class.

.col 1 of 3 (33%)	.col 1 of 3 (33%)	.col 1 of 3 (33%)
.col 1 of 1 (100%)		
.col 1 of 2 (50%)	.col 1 of 2 (50%)	
.col-8 Fixed 8 of 12 (66%), the other columns in the row will split the remaining 4 (33%)	.col	.col

(<https://demo.tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/?2-auto-layout>)

Automatic Layout

3. Column Wrapping

When the sum of all columns in a row is over 12, the first extra column will move to the next line. This is known as column wrapping and works the same way it did in non-flexbox bootstrap.

🔗 (<https://www.facebook.com/sharer/sharer.php?u=https://tutorialzine.com/2016/11/bootstrap-4-regular->

```

<div class="row">
  <div class="col-6">.col-6</div>
  <div class="col-6">.col-6</div>
  <div class="col-3">.col-3, This column will move to the next line.</div>
</div>

```

The only thing to note here is that when using the auto layout, a sizeless column that took up only a couple of spaces, can take up the entire row once it wraps.

col-3	col-3	col In this scenraio .col equals .col-6
col-6		col-6
col Here the column wraps and the same .col class equals .col-12		

(<https://demo.tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/?3-column-wrapping>)

Column Wrapping

4. Responsive Grid

As we mentioned in the intro, Bootstrap 4 has a new XL grid tier on top of the old ones. Now the grid media queries look like this:

- Extra small (xs) - below 576px
- Small (sm) - between 576px and 768px
- Medium (md) - between 768px and 992px
- Large (lg) - between 992px and 1200px
- Extra Large (xl) - over 1200px

Other than that, there haven't been any changes to the way responsiveness works.

🔗 (<https://www.facebook.com/sharer/sharer.php?u=https://tutorialzine.com/2016/11/bootstrap-4-regular->



col-6 col-sm	col-6 col-sm	col-6 col-sm	col-6 col-sm
col-sm	col-sm	col-sm	

(<https://demo.tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/?4-responsive-grid>)

Responsive Grid

5. Column Height

The old grid system was built on floated elements and because of that every column has a different height, depending on the content it holds.

col-xs-4 Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt	col-xs-4 Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam	col-xs-4 Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.
---	---	--

(<https://demo.tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/?5-column-height>)

Column Height In Bootstrap 3

In flexbox layouts all cells in a row are aligned to be as tall as the column with most content.

col-4 Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt	col-4 Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam	col-4 Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
--	--	--



(<https://www.facebook.com/sharer/sharer.php?u=https://tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/?5-column-height>)



(<https://demo.tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/?5-column-height>)

Column Height in Bootstrap 4

6. Horizontal Alignment

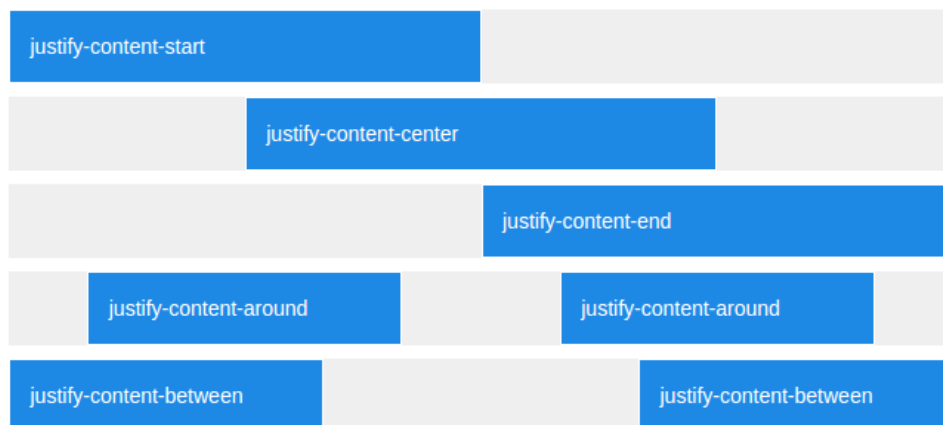
In old Bootstrap, positioning columns horizontally is done via an offset system. Offsets work like empty columns and allow us to move elements to the right (e.g an `.col-xs-offset-3` moves the column 3 spaces to the right). This can be a little annoying as we need to manually adjust the amount of spaces needed.

```
<div class="row">
  <div class="col-xs-6 col-xs-offset-3">This column is now centered.</div>
</div>
```


Thanks to the `justify-content` (<https://developer.mozilla.org/en-US/docs/Web/CSS/justify-content>) property, horizontal positioning in flex-strap is as easy as adding the correct class.

```
<div class="row justify-content-center">
  <div class="col-6">All columns in that row will be automatically centered.</div>
</div>
```

Also, if you want to use offsets, you can still do that as well! Just keep in mind that the classes are now shortened to `.offset-xs-*`.



(<https://demo.tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/?6-horizontal-alignments>)

 (<https://www.facebook.com/sharer/sharer.php?u=https://tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/>)



7. Vertical Alignment

There are no options for vertical alignment in the Bootstrap 3 grid. The only way to do any sort of vertical positioning is using custom CSS and it is often messy.

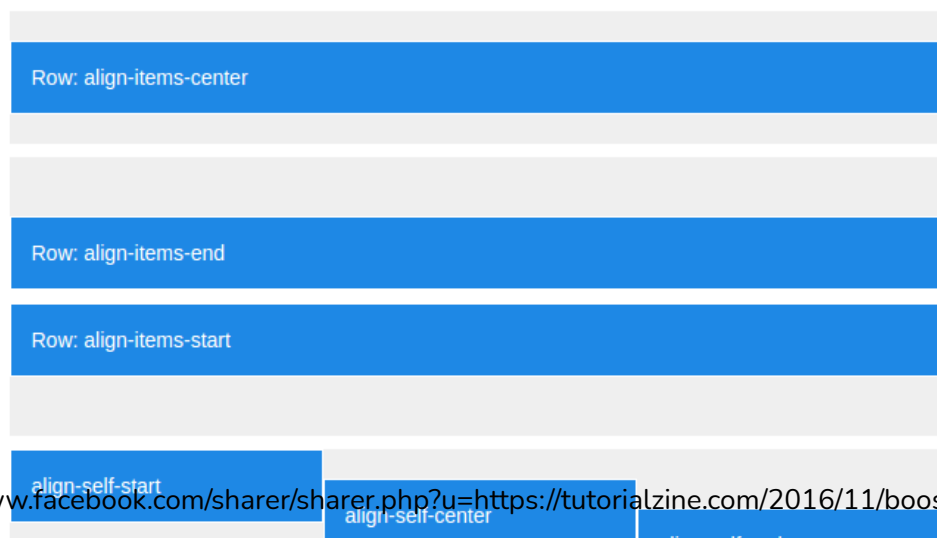
Flexbox, on the other hand, is great at layout alignment and gives us not one, but two ways to vertically position columns:

Vertically align the whole row:

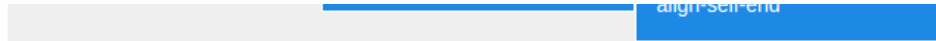
```
<div class="row align-items-center">
  <div class="col">Middle</div>
</div>
<div class="row align-items-end">
  <div class="col">Bottom</div>
</div>
<div class="row align-items-start">
  <div class="col">Top</div>
</div>
```

Align individual columns within the row:

```
<div class="row">
  <div class="col align-self-start">Top</div>
  <div class="col align-self-center">Middle</div>
  <div class="col align-self-end">Bottom</div>
</div>
```



(<https://www.facebook.com/sharer/sharer.php?u=https://tutorialzine.com/2016/11/bootstrap-4-regular->



<https://demo.tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/?7-vertical-alignment>

Vertical Alignment

8. Reordering Columns

With the old grid system, if we wanted to swap around the order of columns we needed to use `push` and `pull` while manually adjusting the correct amount of places to move left and right.

```
<div class="row">
  <div class="col-xs-4 col-xs-push-8"> This column will move 8 spaces to the right
  <div class="col-xs-8 col-xs-pull-4"> This column will move 4 spaces to the left.
</div>
```

Anyone who has used flexbox before knows that it has a built-in `order` (<https://developer.mozilla.org/en-US/docs/Web/CSS/order>) property. The way Bootstrap devs have implemented it is via three ordering classes:

- `.flex-first` - Displayed first.
- `.flex-last` - Displayed last.
- `.flex-unordered` - Displayed between first and last.

No manual calculations are required. If you need to order more than 3 columns (which rarely happens) you can use `push&pull` (<https://v4-alpha.getbootstrap.com/layout/grid/#push-and-pull>) or the `order` (<https://developer.mozilla.org/en-US/docs/Web/CSS/order>) property via CSS.

2nd in markup First in layout <code>flex-first</code>	3rd in markup Unordered in layout <code>flex-unordered</code>	1st in markup Last in layout <code>flex-last</code>
---	---	---

<https://demo.tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/?8-reordering-columns>

Reordering Columns

Conclusion

 <https://www.facebook.com/share/sharer.php?u=https://tutorialzine.com/2016/11/bootstrap-4-regular-vs-flex-grid/>
Looking back at the points covered in the article, it's pretty clear that going full flexbox brings a

lot of great features and makes the grid system much more advanced and versatile. The only real drawback is the lack of support for IE9 and older browsers (all modern clients have full flexbox compatibility (<https://caniuse.com/#feat=flexbox>)), if you don't have to support those you're good to go.

This wraps up our introduction to the new Bootstrap 4 grid system. Feel free to bookmark the demo page for quick future reference. We hope we've been helpful! Happy coding :)

Related Articles

Learn the Bootstrap Grid in 15 Minutes


(<https://tutorialzine.com/2015/10/learn-the-bootstrap-grid-in-15-minutes>)

Bootstrap (<https://tutorialzine.com/tag/bootstrap>) | Quick Learn (<https://tutorialzine.com/tag/quick-learn>)

Creating Tesla's Website With Bootstrap 4

(<https://tutorialzine.com/2018/02/creating-teslas-website-with-bootstrap-4>)

Bootstrap (<https://tutorialzine.com/tag/bootstrap>)

 (<https://www.facebook.com/sharer/sharer.php?u=https://tutorialzine.com/2016/11/bootstrap-4-regular->



CSS Grid VS Flexbox: A Practical Comparison

(<https://tutorialzine.com/2017/03/css-grid-vs-flexbox>)

Flexbox (<https://tutorialzine.com/tag/flexbox>) | CSS Grid (<https://tutorialzine.com/tag/css-grid>)

Comments 13 Oldest ▼

Rilind Pozhegu

2 years ago

Bootstrap 4 is Amazing, I suggest to use it even that's on Alpha version.

Michael F.

1 year ago

don't work demo

Riad Us Salehin

1 year ago

Nicely explained article. However, to get a thorough idea on How the overall Grid Systems Work In Bootstrap 4 (<https://fluentthemes.com/how-grid-systems-work-in-bootstrap-4/>) can read the article which I found pretty useful.

Ferrrmolina

3 years ago

Great article. Thanks!



Maxx Astral (<https://www.facebook.com/sharer/sharer.php?u=https://tutorialzine.com/2016/11/bootstrap-4-regular->

3 years ago

≡ Very nice article on the new Bootstrap 4 Grid systems... Obviously Flexbox Grid is the modern way of doing thing as it lets the engine take care of the calculation nuances...

Khaleel

3 years ago

One thing I have always hated is flexbox. Not supported well in Mobiles or Safari - I have recoded so many client sites because of the lack of support that it now has; that I have for life decided to hate it. The annoying order feature and flex-order and flex width - no thanks. I will stick to divs!

Danny Markov

3 years ago

Actually flexbox is now supported by almost all popular browsers. Here is a link to caniuse (<https://caniuse.com/#search=flex>), and as you can see it's almost all green across the board.

Safari also supports it, just needs prefixes (which yes - annoying), but Bootstrap fixes that for you (yay Bootstrap!).

Fables

3 years ago

Good Article and comparison!
Thanks!

Tarek

3 years ago

Everybody should be using flexbox for layouts, whether in bootstrap or anything else. it's just more powerful with much more control.

Bryan

3 years ago

... unless you need support for earlier browsers!

Khaled Nobani

(<https://www.facebook.com/sharer/sharer.php?u=https://tutorialzine.com/2016/11/bootstrap-4-regular->

Great article, this one is awesome.

Shamim Mia

3 years ago

Wonderful explanation and comprehensive examples! Thank you!

Alex C

3 years ago

Great article, thanks! It's good to see Flexbox being integrated into Bootstrap. It's a really nice way to do layout.

A community that learns together.

Article License (<https://tutorialzine.com/license>)

Terms and Conditions (<https://tutorialzine.com/terms>)



Privacy Policy (<https://tutorialzine.com/privacy>)


Contact Form (<https://tutorialzine.com/contact>)

The Book (<https://tutorialzine.com/books/jquery-trickshots>)

Copyright © Zine EOOD (<http://zine.bg/>). All Rights Reserved.

 (<https://www.facebook.com/Tutorialzine>)

 (<https://twitter.com/Tutorialzine>)  (<https://github.com/tutorialzine>)

 (<https://www.facebook.com/sharer/sharer.php?u=https://tutorialzine.com/2016/11/bootstrap-4-regular->