

CSS Flexbox

Created By	Ⓔ Nate L
Last Edited	@Jan 3, 2019 2:28 PM

- Using the display property and setting it to "flex" turns an element into a flex container
 - You can align children of the container (flex items) in rows or columns
 - By default, they will arrange into rows. You can specify this using the flex-direction property. Options are `row`, `column`, `row-reverse`, and `column-reverse`
 - The justify-content property gives you more control over your flexbox container by letting shift you content along the container's **main axis**
 - For example, you might want to center content on a horizontal axis by setting the flex-direction to row, and justify-content to center
 - The align-items property is similar to justify-content, except it works along the **cross axis**, which is opposite of the main axis
 - The flex-wrap property splits a row or column and has its contents wrap around on the new line. By default, container content does not wrap.
 - The flex-shrink property is applied to items within a container and establishes a size ratio
 - Only activates if the combined size of items exceeds the size of the container itself
 - The flex-grow property is the compliment to flex-shrink
 - The flex-basis property sets the initial size of items before they are potentially modified by flex-shrink or flex-grow
 - The flex property is shorthand for grow, shrink, and basis. It looks like:
`flex: 0 1 auto`
 - Remember, this is applied to items, not the container

```
/* Box 1 will grow to fill extra space at twice the rate of Box 2
when the container is greater than 300px. Box 2 will shrink at twice
the rate of Box 1 when the container is less than 300px.*/

#box-1 {
  background-color: blue;
  flex: 2, 1, 150px;
  height: 200px;
}

#box-2 {
  background-color: orange;
  flex: 1, 2, 150px;
  height: 200px;
}
```

- The order property lets you manually order container items
- The align-self property lets you align individual items (along the cross axis), rather than all items at once