Home / Articles /

What You Should Know About Collapsing Margins

Author
Geoff Graham

29 Comments
Go to Comments

Published
May 22, 2015

Updated
Apr 26, 2017

OK, so some spacing walks into a bar, trips and falls on the floor. The bartender asks if he should be cut off and the spacing replies, "No, I'm just a collapsed margin."

Sorry, dumb joke.

On a more serious note, collapsing margins are real and they can be a real pain in the neck if you're not aware of what they are and how they behave. Here are a couple of scenarios where we'd see these little buggers come into play.

When Vertical Worlds Collide

Collapsing margins happen when two vertical margins come in contact with one another. If one margin is greater than the other, then that margin overrides the other, leaving one margin.

Let's say we had elements stacked on top of each other, one with a bottom margin and one with a top margin:

css

```css
.module {
  display: block;
  width: 100%;
  height: 150px;
}

.module__top {
  margin-bottom: 25px;
  background-color: #f38a6d;
}

.module__bottom {
  margin-top: 50px;
  background-color: #3bbfef;
}
```

If the modules above are placed next each other in the HTML markup, then we might expect there to be `75px` (`25px` from the top module plus `50px` from the bottom module) between them vertically, right?

Well, in this edition of *CSS Survival of the Fittest*, we only get 50 of those pixels. It's like the bigger margin straight up ate the other one and left nothing behind.

Embedded Pen Here

Go ahead and measure. The space between those two modules amounts to the 50px of the `module__bottom` and kicks the margin from `module__top` to the curb.

A natural reaction might be to keep increasing the smaller margin until it makes a difference. But alas, when both margins are positive numbers:

```
bigger margin = total vertical margin
```

Those of you more mathematically minded than the likes of myself may cleverly ask: what about negative margins? The answer is that it does have a significant impact and reduces the margin between the two elements.

```
50px + (-25px) = 25px
```

In English, if one margin is negative, the negative margin is subtracted from the positive margin, reducing the total vertical margin. Here's what that looks like:

Embedded Pen Here

And if both margins are negative? The same rule holds as if both were positive. However, the margin that collapses is the bigger negative of the two rather than the one that is closest to being positive. For example, if one margin is `-25px` and the other is `-50px`, then `-50px` will be the margin.

Another situation you might wonder about is when one vertical margin meets an equal vertical margin. If this were truly "survival of the fittest", then two clashing equal margins should either live together or die together, right? That's a good thought, but what ends up happening is that one continues to get used and one is collapsed. It's sort of like one swallowed the other, though we're not sure which one gets swallowed.

When Parents Ground Their Children

Collapsing margins also occur when the margin of a child element crosses the margin of its parent. Let's consider the following:

css

```css
/* Parent */
div {
  margin: 15px;
}

/* Here come the children */
.red {
  background-color: #ff6b6b;
}

.orange {
  background-color: #ff9e2c;
}

.yellow {
  background-color: #eeee78;
}

.green {
  background-color: #4ecd9d;
}

.blue {
  background-color: #4e97cd;
}

.purple {
  background-color: #6c4ecd;
}
```

In this example, `div` is the parent and each class after is a nested child element. That establishes the parent-child relationship.

Again, our natural inclination might be to expect the total margin to be the sum of the parent and child margins. That's not the case, however, and the child margin will be overridden by the parent margin - at least as long as the child element lives under the parent element's roof. Parents can be so bossy.

But like any parental punishment, there is a way around this by adding something solid to the parent. In this case, adding `1px` of padding allows both margins to be used.

Embedded Pen Here

The same would be true if we added `border-top` to the parent. As long as something solid sits between the parent and child, both margins will be used.

In Conclusion

Do you see how collapsing margins can make things tricky? I personally encounter this on a frustratingly frequent basis when dealing with typography. I'm tempted to add margins to things like `<h1>` and `<h2>` but that often ends up being the source of a lot of headache down the road as vertical margins collide with one another.

I'm interested to know what you all think about collapsing margins. In particular, how do you manage them in a pattern-based system? Harry Roberts offers (http://csswizardry.com/2012/06/single-direction-margin-declarations/) some nice tips of his own. Please share!