Fighting the Space Between Inline Block Elements

Author
Chris Coyier

161 Comments
Go to Comments →

Published
Apr 21, 2012

Updated
Mar 4, 2016

I've seen this come up a couple of times lately on Twitter and then an interesting Dabblet so I figured it would be an important thing to document.

Here's the deal: a series of `inline-block` elements formatted like you normally format HTML will have spaces in between them.

In other words:

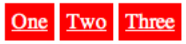
HTML

```html
<nav>
  <a href="#">One</a>
  <a href="#">Two</a>
  <a href="#">Three</a>
</nav>
```

CSS

```css
nav a {
  display: inline-block;
  padding: 5px;
  background: red;
}
```

Will result in:


One Two Three

Often highly undesirable

We often want the elements to butt up against each other. In the case of navigation, that means it avoids the awkward little unclickable gaps.

This isn't a "bug" (I don't think). It's just the way setting elements on a line works. You want spaces between words that you type to be spaces right? The spaces between these blocks are just like spaces between words. That's not to say the spec couldn't be updated to say that spaces between inline-block elements should be nothing, but I'm fairly certain that is a huge can of worms that is unlikely to ever happen.

Here's some ways to fight the gap and get inline-block elements sitting directly next to each other.

Remove the spaces

Remove the spaces

The reason you get the spaces is because, well, you have spaces between the elements (a line break and a few tabs counts as a space, just to be clear). Minimized HTML will solve this problem, or one of these tricks:

HTML

```
<ul>
  <li>
   one</li><li>
   two</li><li>
   three</li>
</ul>
```

or

HTML

```
<ul>
  <li>one</li
  ><li>two</li
  ><li>three</li>
</ul>
```

or with comments...

HTML

```
<ul>
  <li>one</li><!--
  --><li>two</li><!--
  --><li>three</li>
</ul>
```

They're all pretty funky, but it does the trick.

Negative margin

You can scoot the elements back into place with negative 4px of margin (may need to be adjusted based on font size of parent). Apparently this is problematic in older IE (6 & 7), but if you don't care about those browsers at least you can keep the code formatting clean.

css

```css
nav a {
  display: inline-block;
  margin-right: -4px;
}
```

Skip the closing tag

HTML5 doesn't care anyway. Although you gotta admit, it feels weird.

HTML

```html
<ul>
  <li>one
  <li>two
  <li>three
</ul>
```

Set the font size to zero

A space that has zero `font-size` is... zero width.

css

```css
nav {
  font-size: 0;
}
nav a {
  font-size: 16px;
}
```

HEY!

Matt Stow reports that the font-size: 0; technique has some problems on Android. Quote: “Pre-Jellybean does not remove the space at all, and Jellybean has a bug whereby the last element randomly has a tiny bit of space.” See research.

HEY!

Also note, if you're sizing fonts in ems, this zero font size thing can be an issue, since ems cascade the children would also have zero font size. Rems would be of help here, otherwise any other non-cascading `font-size` to bump it back up.

HEY!

Another weirdness! Doug Stewart showed me that if you use @font-face with this technique, the fonts will lose anti-aliasing in Safari 5.0.x. (test case) (screenshot).

Just float them instead

Maybe they don't need to be inline-block at all, maybe they can just be floated one way or another. That allows you to set their width and height and padding and stuff. You just can't center them like you can by `text-align: center;` the parent of `inline-block` elements. Well... you kinda can but it's weird.

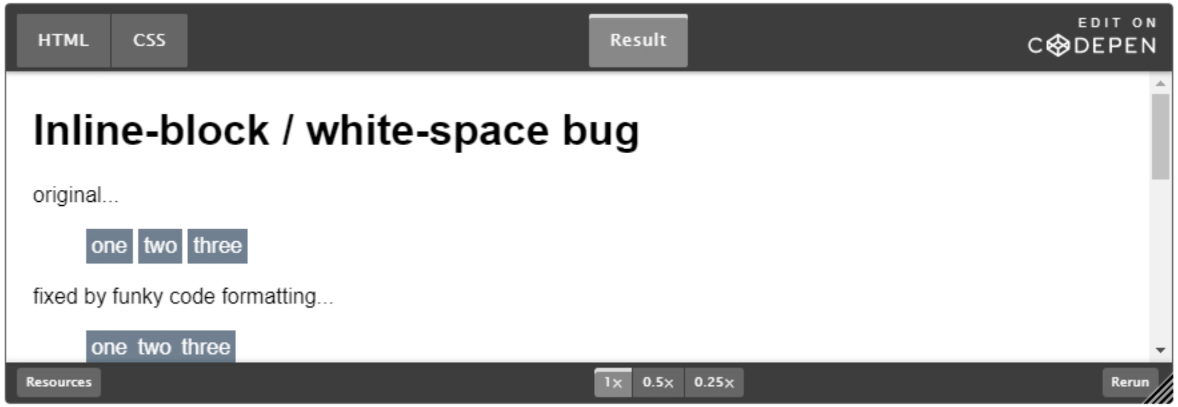
but it's weird.

Just use flexbox instead

If the browser support is acceptable to you and what you need out of inline-block is centering, you could use flexbox. They aren't exactly interchangeable layout models or anything, but you might get what you need out of it.

See

Example on CodePen:



Inline-block / white-space bug

original...

one two three

fixed by funky code formatting...

one two three

Comments

Julian Cizmic

April 21, 2012

I always use the negative margin method, but the floating method, simple as it is, will be of great help to me.

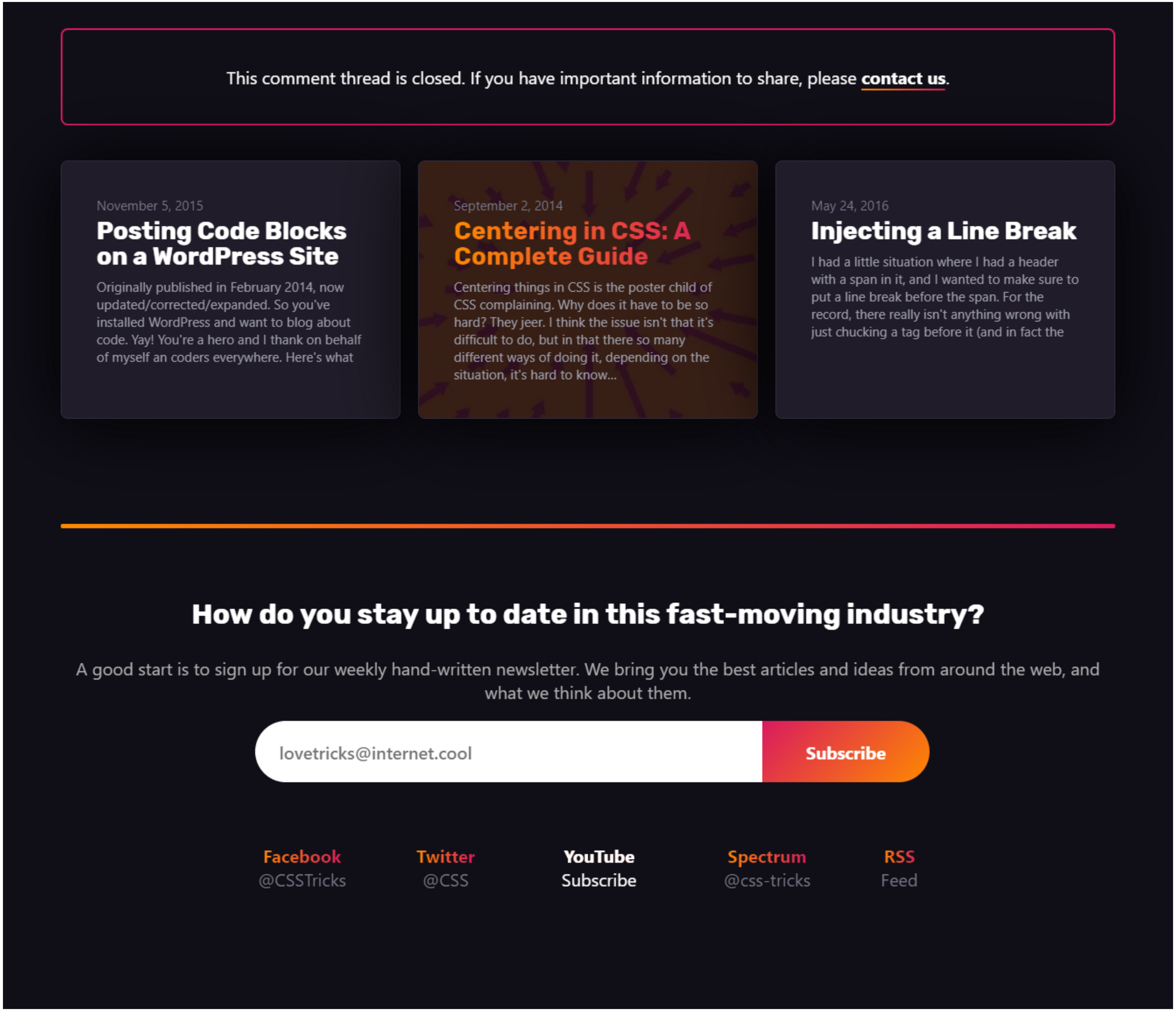
Paolo Tosiani

April 21, 2012

Same here... always negative margin.

That 0px font-size trick looks pretty cool...

Show All Comments



This comment thread is closed. If you have important information to share, please **contact us**.

November 5, 2015

Posting Code Blocks on a WordPress Site

Originally published in February 2014, now updated/corrected/expanded. So you've installed WordPress and want to blog about code. Yay! You're a hero and I thank on behalf of myself an coders everywhere. Here's what

September 2, 2014

Centering in CSS: A Complete Guide

Centering things in CSS is the poster child of CSS complaining. Why does it have to be so hard? They jeer. I think the issue isn't that it's difficult to do, but in that there so many different ways of doing it, depending on the situation, it's hard to know…

May 24, 2016

Injecting a Line Break

I had a little situation where I had a header with a span in it, and I wanted to make sure to put a line break before the span. For the record, there really isn't anything wrong with just chucking a tag before it (and in fact the

How do you stay up to date in this fast-moving industry?

A good start is to sign up for our weekly hand-written newsletter. We bring you the best articles and ideas from around the web, and what we think about them.

lovetricks@internet.cool

Facebook	Twitter	YouTube	Spectrum	RSS
@CSSTricks	@CSS	Subscribe	@css-tricks	Feed

👋

CSS-Tricks* is created, written by, and maintained by **Chris Coyier** and **a team** of swell people. The tech stack for this site is fairly **boring**. That's a good thing! I've used **WordPress** since day one all the way up to **v17**, a decision I'm **very happy with**. I also leverage **Jetpack** for extra functionality and **Local** for local development.

*May or may not contain any actual "CSS" or "Tricks".