

How Do You Do max-font-size in CSS?



Chris Coyier on Jan 29, 2020

CSS doesn't have `max-font-size`, so if we need something that does something along those lines, we have to get tricky.

Why would you need it at all? Well, `font-size` itself can be set in dynamic ways. For example, `font-size: 10vw;`. That's using "viewport units" to size the type, which will get larger and smaller with the size of the browser window. If we had `max-font-size`, we could limit how big it gets (similarly the other direction with `min-font-size`).

One solution is to use a media query at a certain screen size breakpoint that sets the font size in a non-relative unit.

```
body {  
  font-size: 3vw;  
}  
@media screen and (min-width: 1600px) {  
  body {  
    font-size: 30px;  
  }  
}
```

CSS

There is a concept dubbed [CSS locks](https://blog.typekit.com/2016/08/17/flexible-typography-with-css-locks/) (<https://blog.typekit.com/2016/08/17/flexible-typography-with-css-locks/>) that gets fancier here, slowly scaling a value between a minimum and maximum. We've covered that. (<https://css-tricks.com/snippets/css/fluid-typography/>) It can be like...

```
body {  
  font-size: 16px;  
}  
@media screen and (min-width: 320px) {  
  body {  
    font-size: calc(16px + 6 * ((100vw - 320px) / 680));  
  }  
}  
@media screen and (min-width: 1000px) {  
  body {  
    font-size: 22px;  
  }  
}
```

CSS

```
}  
}
```

We've also covered how [it's gotten \(or will get\) a lot simpler \(https://css-tricks.com/simplified-fluid-typography/\)](https://css-tricks.com/simplified-fluid-typography/).

There is a `max()` function in CSS, so our example above becomes a one-liner:

```
font-size: max(30vw, 30px);
```

CSS

Or double it up with a min and max:

```
font-size: min(max(16px, 4vw), 22px);
```

CSS

Which is identical to:

```
font-size: clamp(16px, 4vw, 22px);
```

CSS

Browser compatibility for these functions is [pretty sparse \(https://developer.mozilla.org/en-US/docs/Web/CSS/clamp#Browser_compatibility\)](https://developer.mozilla.org/en-US/docs/Web/CSS/clamp#Browser_compatibility) as I'm writing this, but Chrome currently has it. It will get there, but look at the first option in this article if you need it right now.

Now that we have these functions, it seems unlikely to me we'll ever get `min-font-size` and `max-font-size` in CSS, since the functions are almost more clear as-is.