

# How to set min-font-size in CSS

Asked 7 years, 4 months ago · Active 2 months ago · Viewed 130k times



I want to set a minimum font size to every element in my HTML page.

89

For example if there are elements with font-size less then 12px, then they will change to 12px. But if there are elements with font-size grater then 12px, they will not change.



20



[html](#) [css](#) [font-size](#)

Share Improve this question

edited Jun 3 '14 at 14:15

asked Jun 1 '14 at 21:23

Follow



[nrofis](#)

7,590

9

46

93

## 13 Answers

Active

Oldest

Votes



In CSS3 there is a simple but brilliant hack for that:

245

```
font-size:calc(12px + 1.5vw);
```



This is because the static part of calc() defines the minimum. Even though the dynamic part might shrink to something near 0.

Share Improve this answer Follow

answered Jun 15 '16 at 13:30



[Marc Ruef](#)

2,783

1

9

13

21 fantastic hack! – [user5716144](#) Jul 3 '16 at 7:19

3 I love you. This is awesome. – [Luke Taylor](#) Sep 22 '16 at 0:17

4 @StefanSteiger, you can use media queries for this: (a)media (max-width: 610px)... – [Adam Wallner](#) Nov 13 '16 at 21:58

2 Brilliant! Exactly what I needed. Thank you! – [spedley](#) Dec 18 '16 at 20:38

6 Although this is a nice hack, I'd say this isn't truly a solution as it affects the font-size when the size is above minimum, so you have to put up with that extra 12px (in this case) at all times. – [Tom Wyllie](#) Jul 31 '17 at 11:47



As of mid-December 2019, the CSS4 min/max-function is exactly what you want:  
**(tread with care, this is very new, older browsers (aka IE & msEdge) don't support it just**

64

**yet)**

(supported as of Chromium 79 &amp; Firefox v75)

<https://developer.mozilla.org/en-US/docs/Web/CSS/min>  
<https://developer.mozilla.org/en-US/docs/Web/CSS/max>

Example:

```
blockquote {
  font-size: max(1em, 12px);
}
```

That way the font-size will be 1em (if 1em > 12px), but at least 12px.

~~Unfortunately this awesome CSS3 feature isn't supported by any browsers yet, but I hope this will change soon!~~

**Edit:**

This used to be part of CSS3, but was then re-scheduled for CSS4.

As per [December 11th 2019](#), support arrived in Chrome/Chromium 79 (including on Android, and in Android WebView), and as such also in Microsoft Chredge aka Anaheim including Opera 66 and Safari 11.1 (incl. iOS)

Share Improve this answer Follow

edited Jun 12 '20 at 8:22

answered Aug 28 '14 at 15:19

**Stefan Steiger****70.3k** 63 345 416

7 seriously, why do standards committees always remove the stuff that would actually make my job doable!? – [Michael](#) Jun 25 '18 at 20:19

2 @Michael, the CSS WG removes stuff mostly because lack of implementer's interest or priority. It needs two independent implementations to get a CSS feature in a final W3C Recommendation. There is no point in writing a paper and nobody wants to push it into real world software. So you should ask browser vendors ... – [jj.](#) Jun 6 '19 at 13:42

2 Great find! Can use with css variables to set min/max. Example: `:root { --responsive-font-size-primary: min(2em, max(0.5em, 5vmin)); }` making usage simple: `font-size: var(--responsive-font-size-primary);` – [Corey Alix](#) Dec 30 '19 at 17:05

@Corey Alix: Cool, just noticed support for that has arrived as per December 11th, 2019. Thread with care, all non-Chromium based browsers don't support it just yet. But support for it in Chrome is very cool news. – [Stefan Steiger](#) Dec 30 '19 at 18:01 ✎

1 @Michael: Good news - it's now implemented ! (per mid-December 2019) – [Stefan Steiger](#) Dec 30 '19 at 18:21

Looks like I'm a bit late but for others with this issue try this code

**9**

```
p { font-size: 3vmax; }
```

use whatever tag you prefer and size you prefer (replace the 3)

```
p { font-size: 3vmin; }
```

is used for a max size.

Share Improve this answer Follow

answered Feb 5 '15 at 4:08



CCC

**467** 5 11

- 
- 6 vmax and vmin are actually used to specify values relative to the viewport/page size. The min and max refer to the minimum/maximum viewport not the min or max of the font size - which is what the op was asking. (e.g. if height is larger than width of the viewport than vmax is equal to 100vmax = height of viewport and 100vmin = width of viewport) – [Chaim](#) Sep 9 '15 at 16:28
- 

Use a media query. Example: This is something im using the original size is 1.0vw but when it hits 1000 the letter gets too small so I scale it up

8

```
@media(max-width:600px){
  body,input,textarea{
    font-size:2.0vw !important;
  }
}
```

This site I m working on is not responsive for >500px but you might need more. The pro,benefit for this solution is you keep font size scaling without having super mini letters and you can keep it js free.

Share Improve this answer Follow

answered Oct 18 '15 at 14:14

[zardilior](#)

**2,334** 22 29

- 
- 1 ugh. this ties the element to the resolution of the window, does it not? which means that in the general you would need a separate media query for each element on the page. – [Michael](#) Jun 25 '18 at 20:18
- 

CSS has a clamp() function that holds the value between the upper and lower bound. The clamp() function enables the selection of the middle value in the range of values between the defined minimum and maximum values.

It simply takes three dimensions:

1. Minimum value.
2. List item
3. Preferred value Maximum allowed value.

try with the code below, and check the window resize, which will change the font size you see in the console. i set maximum value 150px and minimum value 100px .

```
$(window).resize(function(){
    console.log($('#element').css('font-size'));
});
console.log($('#element').css('font-size'));

h1{
    font-size: 10vw; /* Browsers that do not support "MIN () - MAX ()" and "Clamp ()"
functions will take this value.*/
    font-size: max(100px, min(10vw, 150px)); /* Browsers that do not support the "clamp
()" function will take this value. */
    font-size: clamp(100px, 10vw, 150px);
}

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.3.1/jquery.min.js">
</script>
<center>
    <h1 id="element">THIS IS TEXT</h1>
</center>
```

[Run code snippet](#)
[Expand snippet](#)

Share Improve this answer Follow

answered May 19 '20 at 13:32



**Bhavik Hirani**

**1,800** 4 24 40

## CSS Solution:

3

```
.h2{
    font-size: 2vw
}
@media (min-width: 700px) {
    .h2{
        /* Minimum font size */
        font-size: 14px
    }
}
@media (max-width: 1200px) {
    .h2{
        /* Maximum font size */
        font-size: 24px
    }
}
```

Just in case if some need scss mixin:

```
///
/// Viewport sized typography with minimum and maximum values
///
/// @author Eduardo Boucas (@eduardoboucas)
///
```

```

/// @param {Number} $responsive - Viewport-based size
/// @param {Number} $min         - Minimum font size (px)
/// @param {Number} $max         - Maximum font size (px)
///                               (optional)
/// @param {Number} $fallback    - Fallback for viewport-
///                               based units (optional)
///
/// @example scss - 5vw font size (with 50px fallback),
///                minimum of 35px and maximum of 150px
/// @include responsive-font(5vw, 35px, 150px, 50px);
///

@mixin responsive-font($responsive, $min, $max: false, $fallback: false) {
  $responsive-unitless: $responsive / ($responsive - $responsive + 1);
  $dimension: if(unit($responsive) == 'vh', 'height', 'width');
  $min-breakpoint: $min / $responsive-unitless * 100;

  @media (max-#{$dimension}: #{$min-breakpoint}) {
    font-size: $min;
  }

  @if $max {
    $max-breakpoint: $max / $responsive-unitless * 100;

    @media (min-#{$dimension}: #{$max-breakpoint}) {
      font-size: $max;
    }
  }

  @if $fallback {
    font-size: $fallback;
  }

  font-size: $responsive;
}

```

Share Improve this answer Follow

edited Jan 10 '19 at 6:14

answered Aug 29 '16 at 8:34



Syed

12.7k

11 93 125



2

No. While you can set a base font size on `body` using the `font-size` property, anything after that that specifies a smaller size will override the base rule for that element. In order to do what you are looking to do you will need to use Javascript.



You could iterate through the elements on the page and change the smaller fonts using something like this:



```

$("*").each( function () {
  var $this = $(this);
  if (parseInt($this.css("fontSize")) < 12) {
    $this.css({ "font-size": "12px" });
  }
});

```

Here is a Fiddle where you can see it done: <http://jsfiddle.net/mifi79/LfdL8/2/>

Share Improve this answer Follow

edited Jun 1 '14 at 21:51

answered Jun 1 '14 at 21:27

mifi79

1,076

6 8

Is there a simple way to find with jQuery elements with size less then 12px? – [nrofis](#) Jun 1 '14 at 21:32

@nrofis - sure, see the edit to my answer above. Cheers! – [mifi79](#) Jun 1 '14 at 21:52

- 11 Iterating over ALL the elements on a page and changing a CSS property seems like a really bad idea from a performance stand point. I'm not an expert on what triggers reflows but you could inadvertently be reflowing the page a lot with this approach. I do not recommend this approach. – [Avand Amiri](#) Sep 20 '14 at 0:16

I'm with @AvandAmiri - this is a bad approach in the sense that this kind of thinking leads to many other solutions like this on a page, which then leads to non-performant sites and even memory leaks that will crash your app. The true answer to this is (until font-size max becomes a legitimately usable thing) is to tighten up architecture - design and implement a good system that just does things properly. There should be no font sizes on a page that are "accidental". – [dudewad](#) Jan 26 '15 at 21:29

- 1 @AndersLindén In that case, set a minimum font size in a media query, for instance, if you don't want your font size to go below 10px: `p{ font-size: 1vw }` `@media (max-width: 1000px){ p{ font-size: 10px }` } – [nHaskins](#) Jul 22 '15 at 20:27



AFAIK it's not possible with plain CSS,  
but you can do a pretty expensive jQuery operation like:

1

[jsBin demo](#)



```
$('.*').css('fontSize', function(i, fs){
  if(parseInt(fs, 10) < 12 ) return this.style.fontSize = "12px";
});
```

Instead of using the *Global Selector* \* I'd suggest you (if possible) to be more specific with your selectors.

Share Improve this answer Follow [edited Jun 1 '14 at 21:58](#)

answered Jun 1 '14 at 21:51



[Roko C. Buljan](#)

**171k** 32 269 282



1

The `font-min-size` and `font-max-size` CSS properties were [removed](#) from the CSS Fonts Module Level 4 specification (and never implemented in browsers AFAIK). And the CSS Working Group replaced the CSS examples with `font-size: clamp(...)` which doesn't have the greatest browser support yet so we'll have to wait for browsers to support it. See example in <https://developer.mozilla.org/en-US/docs/Web/CSS/clamp#Examples>.



Share Improve this answer Follow

answered Feb 12 '20 at 11:32



[Null](#)

**719** 8 11



1

```
.class {
  font-size: clamp(minimum-size, preferred-size, maximum-size)
}
```



using this you could set it up so preferred and max values are 5vw but the minimum is 15px or something so it won't go over 5vw but if 5vw < 15px it will stick to 15px

Share Improve this answer Follow

edited Jan 16 at 14:18

answered Jan 14 at 23:29



**Bobby Mannino**

54 4



Judging by your above comment, you're OK doing this with jQuery — here goes:

0



```
// for every element in the body tag
$("*", "body").each(function() {
    // parse out its computed font size, and see if it is less than 12
    if ( parseInt($(this).css("font-size"), 10) < 12 )
        // if so, then manually give it a CSS property of 12px
        $(this).css("font-size", "12px")
});
```

A cleaner way to do this might be to have a "min-font" class in your CSS that sets font-size: 12px, and just add the class instead:

```
$("*", "body").each(function() {
    if ( parseInt($(this).css("font-size"), 10) < 12 )
        $(this).addClass("min-font")
});
```

Share Improve this answer Follow

answered Jun 1 '14 at 21:51



**Patrick Harrington**

43.2k 5 21 19



A recent feature added to CSS is min() which chooses the minimum value of its parameters.

0



So in the following code, when the viewport is big (>1500px in this case), the font-size is capped to 30px.

However for any viewport smaller than 1500px, the other value is used: e.g. when viewport is 1000px, then the font-size will be 20px. This means it can be quite responsive.

```
p {
    font-size: min(2vw, 30px);
}
```

Here's a Codepen with min() for container width and p text:

<https://codepen.io/code0312/pen/dyWJLmB>

But note, if you later manually set a font size greater than what you put here, that will still override this declaration.

Share Improve this answer Follow

answered Jul 26 at 18:00

**James**



11 2



It will work perfectly with 50px. Which will act as a static and thus as min-width.

-1

```
font-size: calc(50px + 5vw);
```



Share Improve this answer Follow

edited Jun 2 '20 at 20:25



cela

2,110 3 20 37

answered Jun 2 '20 at 18:47



Jayant Khandelwal

7 1