

EdX and its Members use cookies and other tracking technologies for performance, analytics, and marketing purposes. By using this website, you accept this use. Learn more about these technologies in the [Privacy Policy](#).



[Course](#) > [Modul...](#) > [4.2 The...](#) > [Elemen...](#)

Audit Access Expires Apr 27, 2020

You lose all access to this course, including your progress, on Apr 27, 2020.

Upgrade by Apr 13, 2020 to get unlimited access to the course as long as it exists on the site. **Upgrade now**

Element width and height

Element width and height

Until now we've let the browser decide how big the element is, but you can actually adjust its width and height manually.

Width and height

Documentation: [the width property](#) and [the height property](#).

```
1. p {  
2.     width: 30%;  
3. }
```

You can use pixel values for both width and height, but you'll most often want to use percentages to set these so that your elements grow and shrink as appropriate based on the screen size.

For example, if we set the width of a paragraph to 30% as you resize the browser window, you'll see how that element dynamically resizes. That's because when you use percentages, the size is computed based on the element's "containing block", or the element that contains the one you're styling. If your element is just within the body tag, the width is computed based on the relationship with the screen width.

Things are a bit more complicated with using a percentage to set an element's height. This is because typically the body's height is not specified, so if you use a percentage the size won't adjust.

min-width, max-width, min-height, max-height

Documentation: [max and min width](#) and [max and min height](#)

Setting width and height with percentages will save you work because your design will automatically optimize for the user's screen size. However, some elements can't grow and shrink as dynamically as text can.

For example, images will get "pixelated" if you let them grow too large, and they can look really distorted. Thankfully, you can set max and min width and heights. This way, you can set a range for your image to grow and shrink where you know it will still look good.

```
1. img {  
2.     width: 100%;  
3.     max-width: 1024px;  
4. }
```

HTML	CSS	Result	EDIT ON
<pre><!DOCTYPE html> <!--It's a best practice to always declare DOCTYPE!--> <html lang="en"> <head> <title>Dynamic Width and Height</title> <meta charset="utf-8"> </head> <body> <p id="width50"> width 50%, no min or max</pre>		<p>width 50%, no min or max</p> <p>width 80%, max of 1000px</p> <p>width 30%, min of 200px</p> <p>the height is set to 25%, but there is no affect</p> <p>this height is also set to 25%, and since it's within a div containing block who has a set height of 500px this p will resize in relation to that</p>	
Resources		1x 0.5x 0.25x	Rerun

When you view the above example, the paragraphs will dynamically resize based on the size of your window. For example, here is what the code looks like in a wide window:

width 50%, no min or max

width 80%, max of 1000px

width 30%, min of 200px

the height is set to 25%, but there is no affect

this height is also set to 25%, and since it's within a div containing block who has a set height of 500px this p will resize in relation to that

However, here is the exact same code viewed in a much narrower window:

width 50%, no
min or max

width 80%, max of 1000px

width 30%, min of 200px

the height is set to 25%, but there is
no affect

this height is also set to 25%, and
since it's within a div containing
block who has a set height of 500px
this p will resize in relation to that.

Here, you can see that the elements have resized accordingly, but have hit the limits of their min and max constraints. This is why using percentages for width and height are so important, it helps you write code that works for all screen sizes.

[Learn About Verified Certificates](#)

© All Rights Reserved