Web Dev Simplified Blog

Preferences

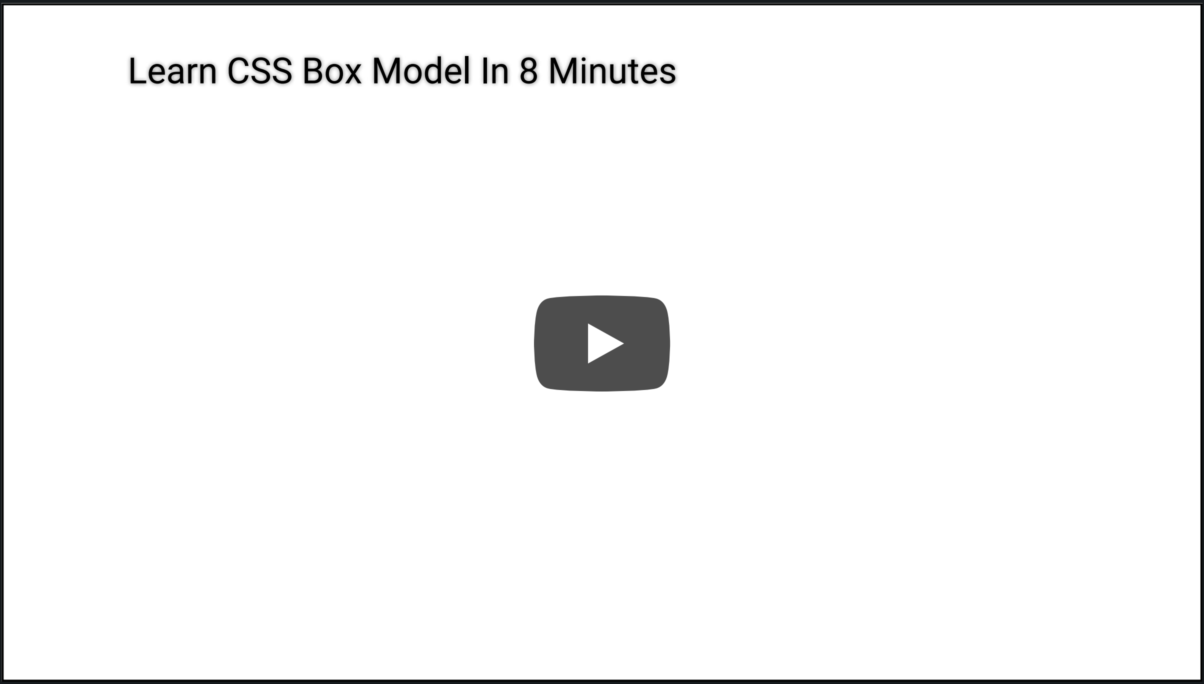
CSS Box Model

December 20, 2021

CSS

Sizing and spacing elements in CSS is confusing. You can do spacing with margin and padding and there are multiple different ways that sizes are calculated in CSS. All of this can be quite confusing to wrap your head around, but luckily everything related to sizing and spacing in CSS can be learned by simply understanding the box model in CSS. Also, I have a complete CSS box model cheat sheet that covers all the concepts from this article plus more with visual examples that you can download for free.

If you prefer to learn visually, check out the video version of this article.



What Is The Box Model?

One incredibly important thing to understand about CSS is that everything is a rectangular box. Everything from text, to buttons, to inputs are represented by a box in CSS. Even if you draw a circle with CSS it will still be represented by a box when it is drawn on screen. This box is composed of multiple parts which in total make up the box model.

Content

The content of a box is everything inside the box. This is usually some form of text or a hard coded width and height. Below we have a simple div and everything with the red background color is the content of the box.

```css
div {
  background-color: red;
  height: 100px;
  width: 200px;
}
```

This red area is exactly 200px wide and 100px tall because of the width and height we added to the div.

Padding

The next portion of the box model is the padding and this comes directly outside the content. Padding allows you to add extra space to your eleme without making the content itself larger. It also maintains the background color of the element. This is perfect for when you need to add space between the edge of your background and your element content like with a button.

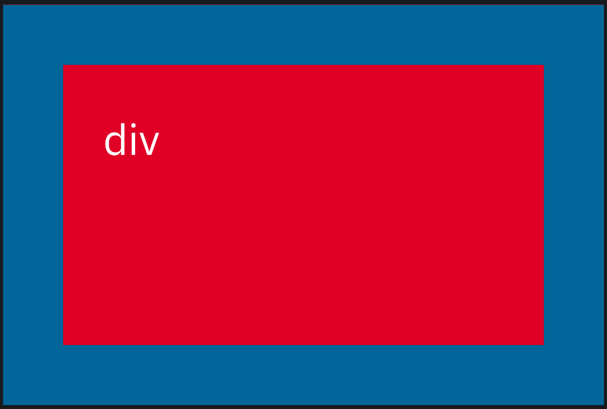
```css
div {
  padding: 20px;
  background-color: red;
  height: 100px;
  width: 200px;
}
```

The above element now has an extra 20px of space on all sides between the content and the end of the background. You will also notice our box is actually 240px wide and 140px tall. This is because the size of a box in CSS by default is a combination of the content, padding, and border and in this example we have a 200px wide box and 20px of padding on the left and right leading to 240px of total size.

Border

The next layer of the box model is the border and the border is the layer that goes between the padding and the outside of the element. It is also the last part of the box model that contributes to the overall size of a box.

```
div {
  border: 30px solid blue;
  padding: 20px;
  background-color: red;
  height: 100px;
  width: 200px;
}
```

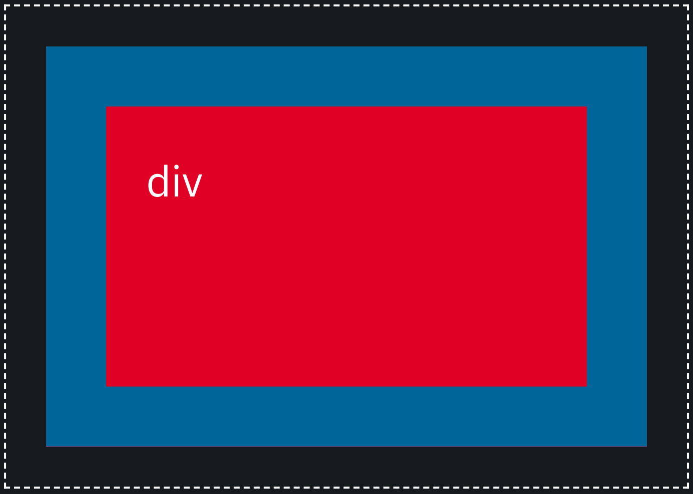


As you can see we now have a 30px blue border around our entire element. The entire size of our element is also 60px wider and taller because of the border. Our overall width is 300px and the height is 200px now.

Margin

The final piece of the box model is margin. Margin is simply the space between different elements and occurs outside the border. Margin also does not contribute to the actual size of an element since it is outside the border.

```
div {
  margin: 20px;
  border: 30px solid blue;
  padding: 20px;
  background-color: red;
  height: 100px;
  width: 200px;
}
```



In the above example I added a container around our element which is represented by the dashed border. As you can see our element is 20px away from the dashed border on all sides because of the margin we applied. You will also notice our element itself is not any larger. This is again because margin does not contribute to the size of an element in the box model.

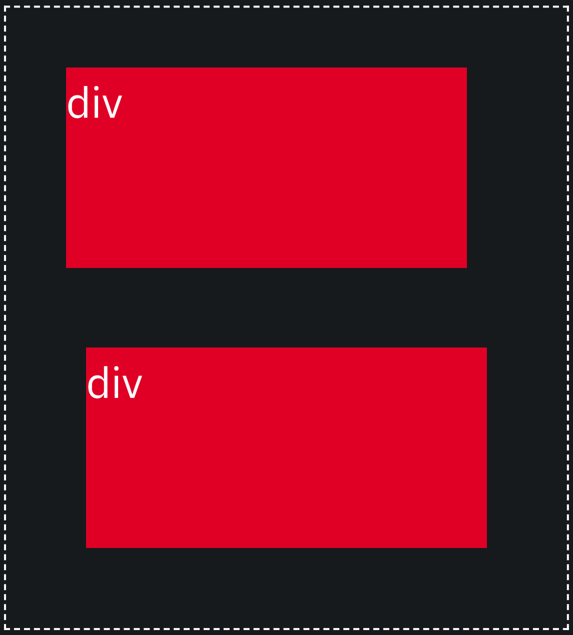
Now these 4 components sum up the entirety of the box model but there are a few gotchas related to how the box model works that you need to understand.

Margin Collapse

In the box model margins between two different elements will collapse. This means that if two elements that are siblings in the HTML both have a marg they will collapse so that only the largest of the two margins is used between the elements. Let's look at a simple example.

```css
.top-div {
  margin: 30px;
  background-color: red;
  height: 100px;
  width: 200px;
}

.bottom-div {
  margin: 40px;
  background-color: red;
  height: 100px;
  width: 200px;
}
```



You will notice that in the above example there are only 40px of space between the top and bottom div. This is because the margins are collapsing and only the largest of the two margins is used as space between these two elements.

This is important to know since it is not always expected behavior so if you have two elements that have a margin and are siblings with each other you need to make sure you keep this in your mind.

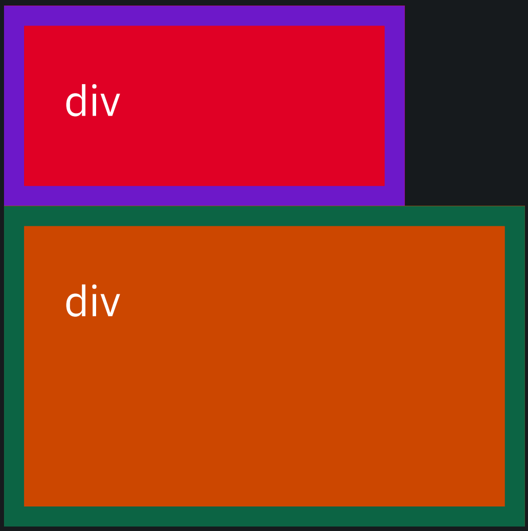
Also, this only works within a normal document flow. If you are using something like flexbox, grid, or another layout method other than the default in CSS then margins do not collapse. This can get really confusing since you may expect them to collapse, but they only collapse in the default layout mode of CSS.

Box Sizing

One of the most important properties in CSS, `box-sizing`, can be used to change how the box model calculates sizes. By default the `box-sizing` of all elements is set to `content-box` which means the content of the element will be the exact width/height defined by the width and height properties in CSS. An alternative value is the `border-box` value which means the width/height properties in CSS will represent the width/height of the content combined with the border and padding.

```
.top-div {
    box-sizing: border-box;
    background-color: red;
    height: 100px;
    width: 200px;
    padding: 20px;
    border: 10px solid purple;
}

.bottom-div {
    box-sizing: content-box;
    background-color: orange;
    height: 100px;
    width: 200px;
    padding: 20px;
    border: 10px solid green;
}
```



The top and bottom div both have the same height/width in CSS, but the bottom div appears much larger. This is because the content of the div is 200px wide in the bottom div while in the top div the content + padding + border is 200px wide. The top div content is actually only 140px wide since 40px of space is taken up by the padding and 20px is taken up by the border

This property may seem a bit useless at first, but being able to set the height/width of an element and having that size include the entire content, border, and padding is incredibly useful. This is why most stylesheets will change the box sizing for all elements.

```css
*, *::before, *::after {
  box-sizing: border-box;
}
```

Conclusion

The box model in CSS is one of the most important concepts you can learn since it encompasses everything about the CSS sizing/layout model into one cohesive idea. Lastly, I have a complete box model cheat sheet that you can download for free to really master everything you need to know about the box model.