

The Box Model

We know that all elements have a rectangular box around them. Each box also has a width and height, but it also can have padding, borders and margins as well. Let's look at all of these parts called the box model.

We've covered a bit already with the width and height of elements.

Every element has a default width. Remember, block-level elements have a default width of 100% and inline and inline-block elements will expand and contract to accommodate their content.

The default *height* of an element is determined by its content, it will expand and contract vertically as necessary to accommodate its content.

Let's look closely at padding, border and margin. The padding and margin properties are very helpful in adding space between various items on the page. This is often referred to as white space (or space without images or text).

Padding allows us to set an amount of space inside the elements border. For example, our `<p>` element that we have here might look something like this with very little padding with the text right up to the edge. It is expanding just wide enough to include the content and not a lot more. We've given the `<p>` element a background color that lies within the border of the element. Padding will allow us to give that paragraph's text some breathing room, or white space. But this extra space is still inside the border of paragraph element with padding. The content is still the same width, but 40 pixels of padding has been added around it inside its border.

Every box has a border (even if it's not showing as is our paragraph—or in other words it has a default width of 0 pixels. The border separates the padding and the margin. If we want a *visible* border it needs three values: width, style and color. For example, `border: 5px solid black`. You can give it any color or width and there are different styles like (double, dashed and dotted). We can even round the corners with `border-radius`, such as `border-radius: 5px`. You can even make a circle with `border-radius: 50%` if your image is a square. Let's apply that one to our image our image was a little wider than it was tall so it makes an oval.

If we want space around our elements, or beyond the border, we can give them margining. The margin allows us to set the amount of space surrounding an element. It's the gap between the borders of two adjacent boxes. Margin spacing will fall outside of the border. It helps to keep elements a certain distance from other elements. Notice how now there is 30 pixels between the top and bottom of the image and paragraph.

The browser default CSS might give a little margining or padding for certain elements but we can always change that through CSS.

Sometimes there is no background color and no visible border so padding and margins seem to do the same thing. But it's important to understand how the box-model works with every element.

It's also helpful to know the longhand and shorthand ways to give values to our padding and margins.

If one value is given that value is applied to all four sides of the element. If we give two values the first value is for the top and bottom and the second value is for the right and left. (padding: 5px 40px); (We saw that with margin: 0 auto.) To set unique values for all four sides you can specify 4 values and they will apply in order from top, right, bottom, then left, clockwise. (Padding: 5px 60px 15px 20px;)

We could also just apply padding or margin to just one side by adding -top, -bottom, -right, or -left, to margin or padding.

It's important to understand just how much room elements really take on our page. By default, when we say the width of an element is 500px for example that doesn't include the padding and border. If we had 20px of padding and a border of 10px on every side the actual full width of the element is 560px. The 500 for width plus 20 padding and 10 border on both sides. The box model can be changed however to calculation the padding and border into the overall width with box-sizing property. For example, box-sizing: border-box. This is different from the default of content-box where border and padding are not included.

So, if I wanted to have my paragraph with its padding and border to be the same width as the image I can use: box-sizing: border-box to do that.