Mastering CSS with Mark Lassoff

Section 5: The CSS Box Model Revisited

Written by Russ Eby and Mark Lassoff

The box model concept is essential for web developers and designers to understand. Everything that appears on the screen on the web is inside a box. The box may go all the way across the screen, it may sit on a baseline of the text, or its width may be limited by CSS. Because of this, understanding the model of how the box works is very important.

HTML

Here is a simple HTML fragment. There is a single logical division (or `div`) and two styles. Our `div` is assigned a background color of red and a text color of white.

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
<head>
    <meta charset="UTF-8">
    <title>Box Model</title>
    <style>
        #box1 {
            background-color: red;
            color: white;
        }
    </style>
</head>
<body>
    <div id="box1">The quick brown fox jumped over the lazy dogs.</div>
</body>
</html>
```

Our box model begins with the inner portion of the box, the content. By default, the width of our box is defined as the width of the content inside it. At this point the width of `#box1` is determined by the space needed to display the text, “The quick brown fox jumped over the lazy dogs.”

> Interesting fact: The sentence “The quick brown fox jumped over the lazy dogs” uses every letter of the alphabet.

Width

By setting the CSS `width` rule for the box, the element will be 50% of the width of the parent element, no matter how wide the window is.

> When width is assigned using a percentage we're using `relative` measurements. When is width is assigned in pixels, we're using an `absolute` measurement. You're likely to run into these terms again when working with CSS elements.

```
box1 {
    background-color: red;
    color: white;
    width: 50%;
}
```

Here is an example of `absolute` width. At 200px the text is too long for the box but the text will flow to accommodate the text, and the height will adjust.

```
box1 {
    background-color: red;
    color: white;
    width: 200px;
}
```

We'll change the width to 400px so that the text now appears on a single line.

Height

The `height` CSS rule allows us to specify the height of the box, regardless of the amount of content inside the element. By setting the height to 400px we now have a 400px by 400px logical division.

```
box1 {
    background-color: red;
    color: white;
    width: 400px;

    height: 400px;
}
```

Padding

If we were to look at the document in the browser, the text is way too close to the edges of the `div` making it difficult to read. By adjusting the `padding`, we create some buffer space between around the text. This space takes on the same background color as the content.

```
#box1 {
    background-color: red;
    color: white;
    width: 400px;
    height: 400px;

    padding: 10px;
}
```

In the example above we have put 10px `padding` around all four sides of our box. We can also control the padding on each side separately.

```
#box1 {
    background-color: red;
    color: white;
    width: 400px;
    height: 400px;

    padding-top: 20px;
    padding-left: 40px;
}
```

Time-Saving Tip

We can also adjust the padding on all four sides of `div` using the following shorthand:

```
box2 {
    padding: 10px 20px 30px 40px;
}
```

The shorthand adjusts the padding in the box starting at the top and continuing in a clockwise direction. So this example would be set the top padding to 10px, the right side to 20px, the bottom to 30px and left side to 40px.

We can also set padding use a slight variation on the shorthand above:

```
box2 {
    padding: 10px 20px;
}
```

If only two values are listed the first number is for top and bottom; the second number applies to the left and right.

The padding adds to the overall width of the box. If you had 40px of padding on the left and right-hand side of the box, the total width of the 400px `div` would be 480px.

Pixel Math

```
#box1 {
    width: 400px;
    padding: 40px;
}
```

At first this may seem like 400px + 40px would result in a width of 440px for the `div`. You have to remember to take in to account the padding on the left **and** right sides resulting in an actual width of 480px.

Margin

The margin is the space between the element and the next element or the parent element. It will take on the background color of the parent element.

```
#box1 {
    background-color: red;
    color: white;
    width: 400px;
    height: 400px;
    padding: 40px;

    margin: 40px;
}
```

Dev Tool Tip:

> Most browsers have a set of dev tools that can be very useful for visualizing the box model.
>
> For Chrome Browsers click on the three dots in the upper right-hand corner and go to Developer tools. When the dev tools open, go to the Elements tab. Keyboard short cut for Chrome is `Control+Shift+C` or `Command+Option+C`.
>
> On the right you will see the box model visualization for whatever element you have highlighted in the code.

Border

According to the box model, between the padding and the margin, going all the way around the element, is the border. The border width and other properties can be selected in your CSS.

```
#box1 {
    background-color: red;
    color: white;
    width: 400px;
    height: 400px;
    padding: 40px;
    margin: 40px;

    border: 20px solid black;
}
```

`border` works a bit differently because it takes three arguments. The example above specifies:

- The width: 20px
- The type: solid
- The color: black

The border also contributes to the total width of the element. , in this case, it would be:

Width	Type
400	Width of content
80	40 for left and right padding
40	20 for left and right border
80	40 for left and right margin
600	**Total width of the logical Division**

Types of borders

- none - This is the default
- hidden - Used mainly for tables where borders are right next to each other
- solid
- dotted
- dashed
- double
- groove
- ridge
- inset
- outset

Submit this

1. Create a Lithuanian flag using three divs.

- Each stripe of the flag will be 400px by 80px
- Top = yellow
- Middle = green
- bottom = red

2. Create a Romanian flag

- Create a div 240px height by 133px width, background is yellow
- Set the top margin to be 100px and the left margin to be 50px
- Add a solid blue 133px left border
- Add a solid red 133px right border

Tip: Yes, the Romanian flag is a single div.

> Written with StackEdit.