HTML5 Authoring with Mark Lassoff

Section 14: Box Model

After completing this section, you will:

- ☐ Identify the Components of the box model
- ☐ Know how to add content in the box model.
- ☐ Understand how to adjust width and height
- ☐ Make adjustments using padding, border, and margin.

Introduction

In web development, the box model is the foundation of layout on the web page. The box model describes the content and the size of each block by styling it through CSS. It is useful to think of your web page as a series of boxes that are stacked, floated and positioned to create the overall layout of the page.

Box Model

As always, we'll begin with the basic document structure.

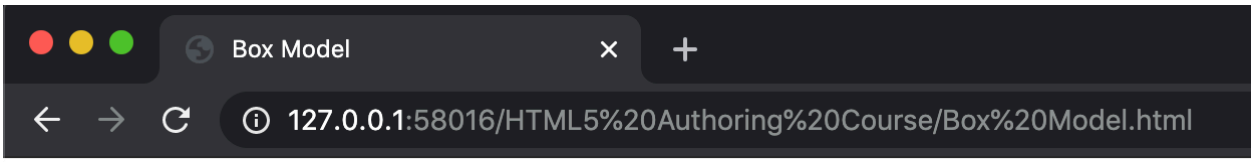
```
<!DOCTYPE html>
<html>
   <head>
      <title>Box Model</title>
   </head>
   <body>

   </body>
</html>
```

Developer Tools

Let's create two logical divisions with unique `id` attributes named `boxA` and `boxB`. We'll also add some text to each `div` tag as content to demonstrate the box model.

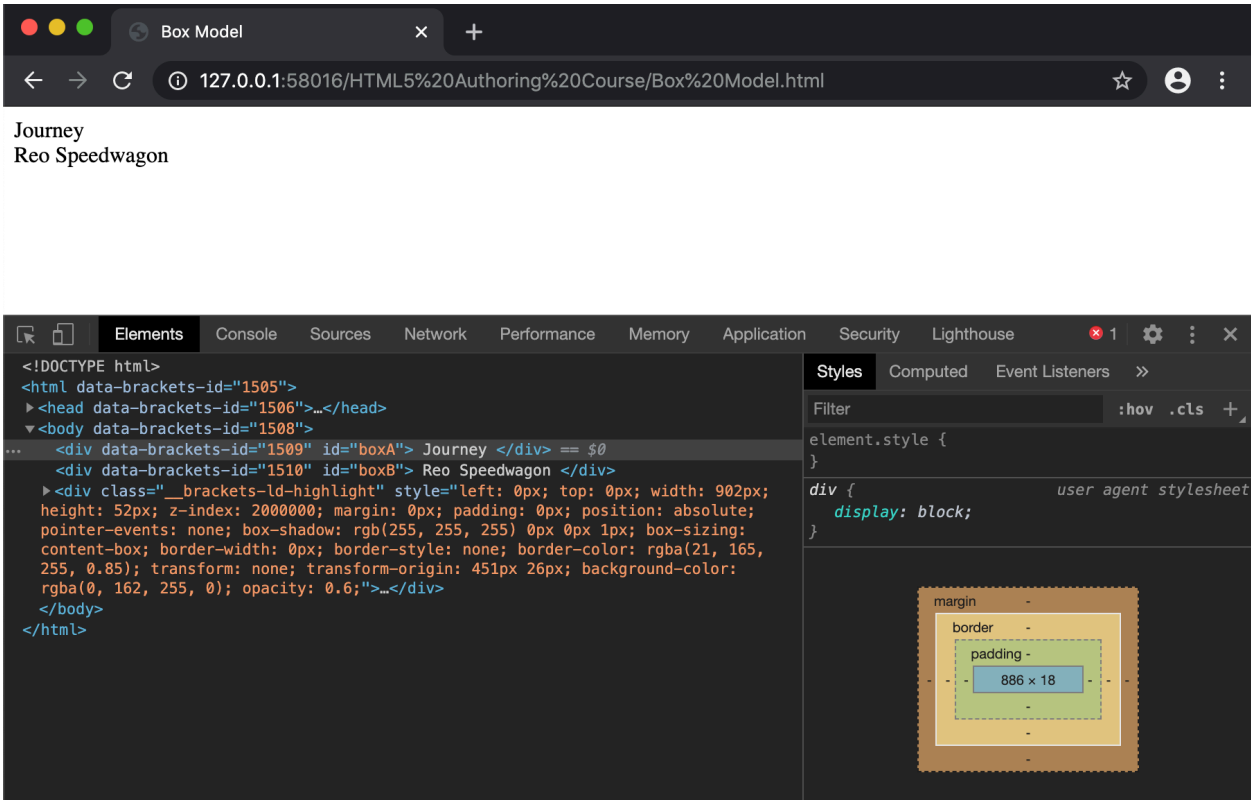
```
<!DOCTYPE html>
<html>
  <head>
    <title>Box Model</title>
  </head>
  <body>
    <div id="boxA"> Journey </div>
    <div id="boxB"> Reo Speedwagon </div>
  </body>
</html>
```

Now, as you can see from the screenshot below, the actual boxes are difficult to discern.



Journey
Reo Speedwagon

Using the developer tools, you can able to view the box around every element on a web page. To open the developer tools in Google Chrome, click `View` > `Developer` > `Developer Tools` and navigate to the Elements tab.



In the Elements tab, you can view all the elements on the web page. The selected element's box should appear in the Computed tab on the top of the rightmost column.

Do This

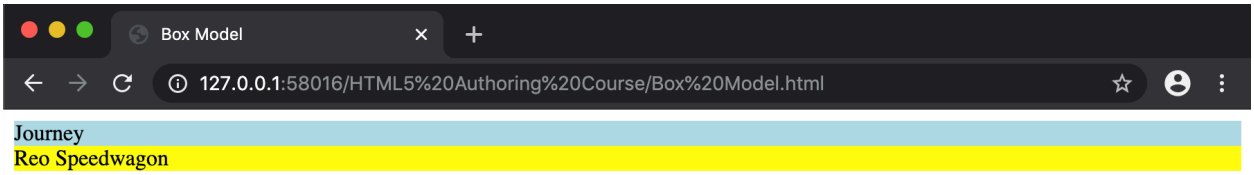
Point at each element in the Developer Tools and watch how your browser reacts in the browser window and the Computed Tab of the Developer tools themselves. Familiarize yourself with the Style tab by viewing it with each of your Div's selected.

Content Box

Let's add a background color to more easily view the content box in our box model.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Box Model</title>
    <style>
      #boxA {
        background-color: lightblue;
      }
      #boxB {
        background-color: yellow;
      }

    </style>
  </head>
  <body>
    <div id="boxA"> Journey </div>
    <div id="boxB"> Reo Speedwagon </div>
  </body>
</html>
```

In the browser, the result of the code above appears like this:



You can see by default our boxes go all the way across the browser window and are as tall as the content inside them.

Next, we'll add more content using `p` tag.

```
<!DOCTYPE html>
<html>
  <head>
```

```
    <title>Box Model</title>
    <style>
      #boxA {
        background-color: lightblue;
      }
      #boxB {
        background-color: yellow;
      }

    </style>
  </head>
  <body>
    <div id="boxA"> Journey
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
    <div id="boxB"> Reo Speedwagon
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
  </body>
</html>
```

Box Height and Width

Several properties determine the size of the box. The `height` and `width` of the element define the core of the box model.

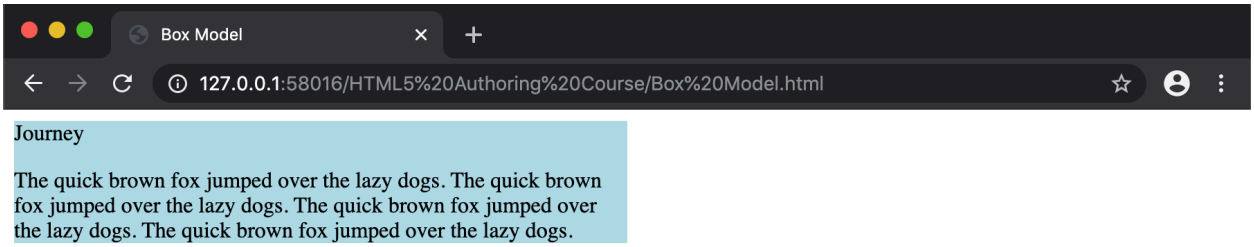
Now, the first thing we will do is to declare a width. Keep in mind that, by default, the width set to 100%. There are a couple of ways we can set the width of an item in HTML. We can set an absolute width in pixels or a percentage with by using a `%` sign.

Keep in mind that many users will be viewing your content on a mobile screen and that by using an absolute width, you may make the site unusable for many.

Let's make a few changes to the code:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Box Model</title>
    <style>
      #boxA {
        background-color: lightblue;
        width: 50%;
      }
      #boxB {
        background-color: yellow;
      }

    </style>
  </head>
  <body>
    <div id="boxA"> Journey
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
    <div id="boxB"> Reo Speedwagon
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
  </body>
</html>
```

You'll notice that the `boxA` is now half of the width of the `boxB`.

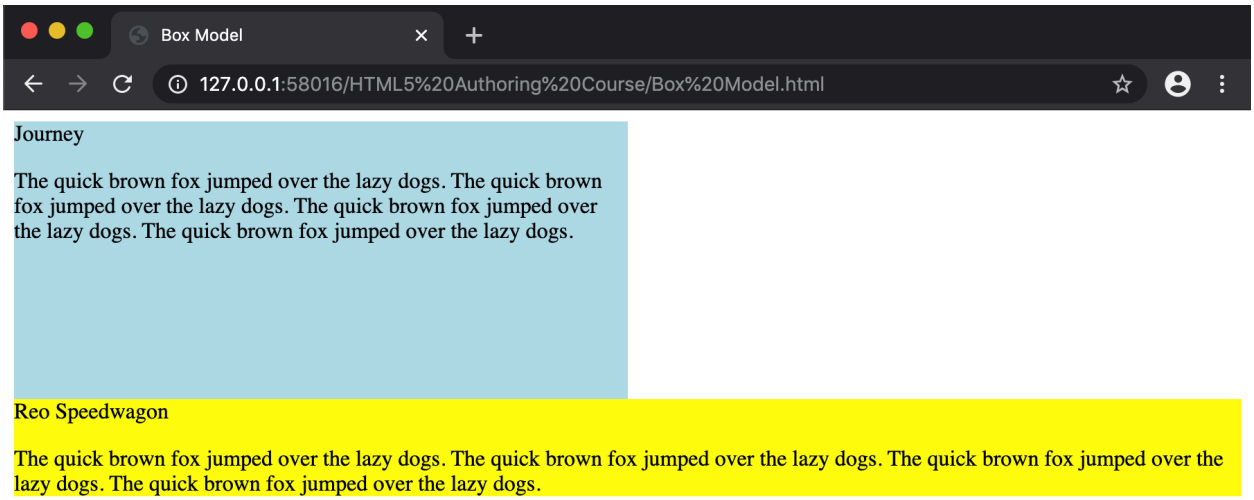


Next, let's adjust the box's height to 200 pixels.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Box Model</title>
    <style>
      #boxA {
        background-color: lightblue;
        width: 50%;
        height: 200px;
      }
      #boxB {
        background-color: yellow;
      }

    </style>
  </head>
  <body>
    <div id="boxA"> Journey
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
    <div id="boxB"> Reo Speedwagon
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
```

```
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
  </body>
</html>
```

Here's the output. Width can be a bit tricky as the width of the box has to accommodate the content, or you'll get unexpected results.



With both width and height set, the box element has taken the desired shape.

Padding

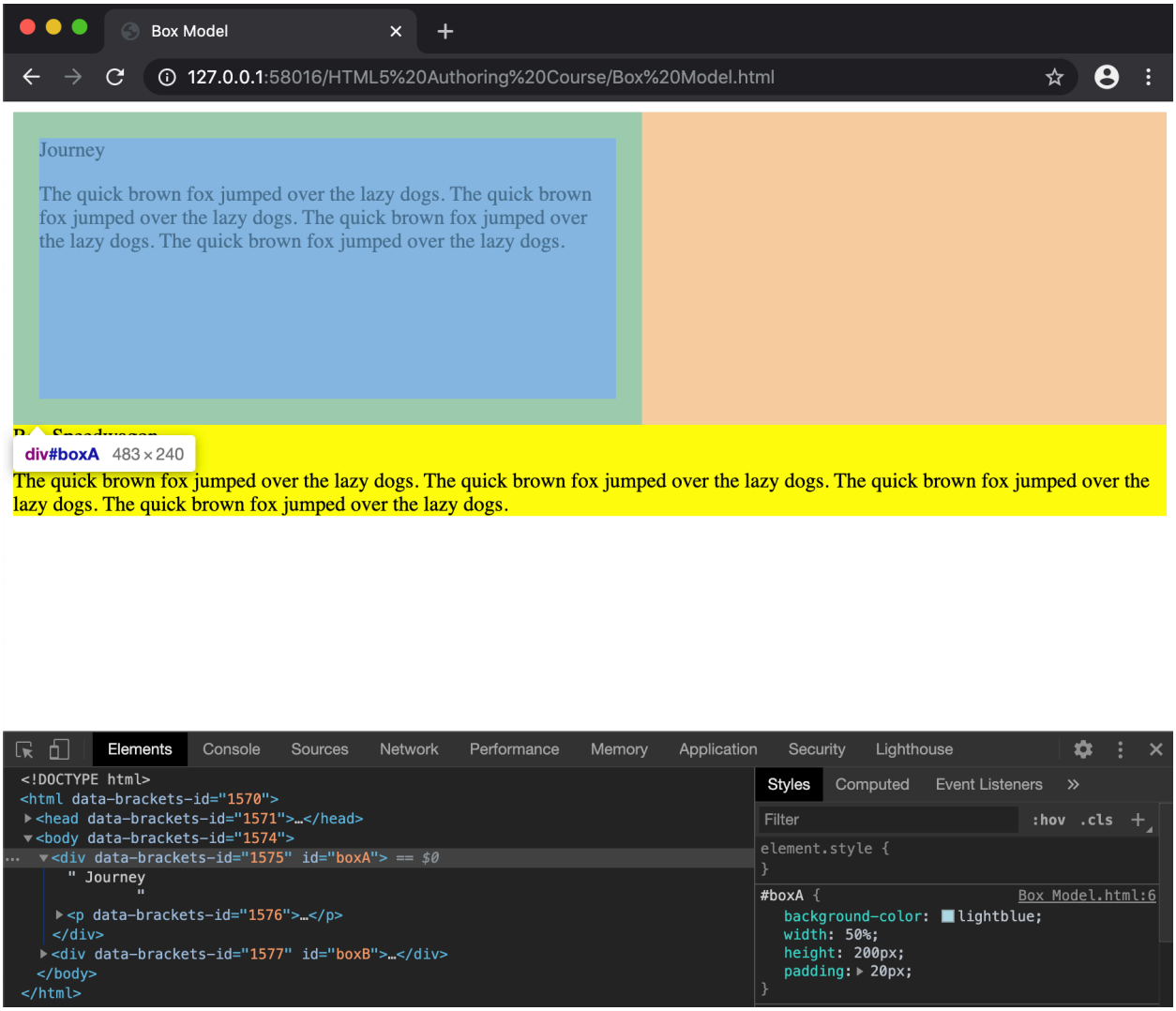
Padding signifies the space between an element's border and the content within.

Here's how we'll use it.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Box Model</title>
    <style>
```

```
        #boxA {
          background-color: lightblue;
          width: 50%;
          height: 200px;
          padding: 20px;
        }
        #boxB {
          background-color: yellow;
        }

    </style>
  </head>
  <body>
    <div id="boxA"> Journey
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
    <div id="boxB"> Reo Speedwagon
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
  </body>
</html>
```

Now, the contents are pulled away from the side of the box.



As you can see, we used shorthand, and by setting only one value for padding, the padding was set on all sides of the box. Typically the pattern `padding: top right bottom left`; would be used.

You may also use individual style rules for each side of the box by using `padding-top`, `padding-right`, `padding-bottom`, and `padding-left` and setting individual rules for each.

Border

The next layer of the box model is `border`. Borders exist around every HTML element, even if you don't see them. The border can take on different sizes, styles, and colors.

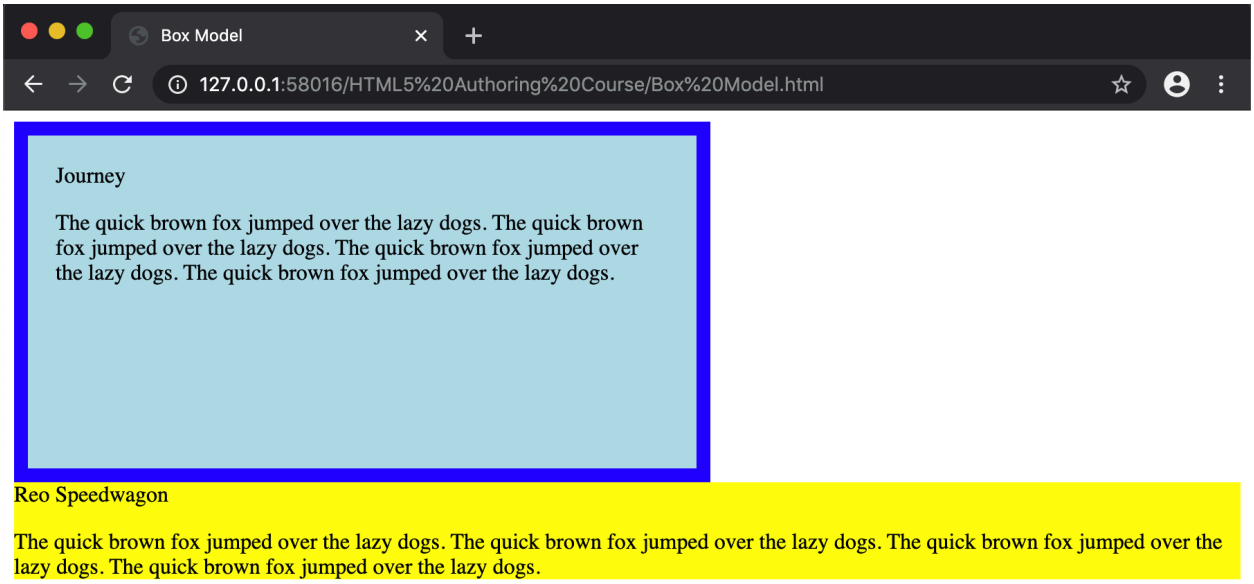
Let's add border with the following code:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Box Model</title>
    <style>
      #boxA {
        background-color: lightblue;
        width: 50%;
        height: 200px;
        padding: 20px;
        border: 10px solid blue;

      }
      #boxB {
        background-color: yellow;
      }

    </style>
  </head>
  <body>
    <div id="boxA"> Journey
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
    <div id="boxB"> Reo Speedwagon
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
```

```
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
  </body>
</html>
```

The CSS border rule has three values: `width`, `style`, and `color`. Width is the thickness of the border (usually in pixels). Style can be `dashed`, `solid`, `dotted` or other patterns. Color is declared using the hex value, an RGB value like `rgb(200,110,92)` or color name like `blue`.

With the blue border our box will look like this when displayed on the page:



Keep in mind that it's also possible to set different border values on different sides of an element via `border-top`, `border-right`, `border-bottom`, and `border-left`.

Margin

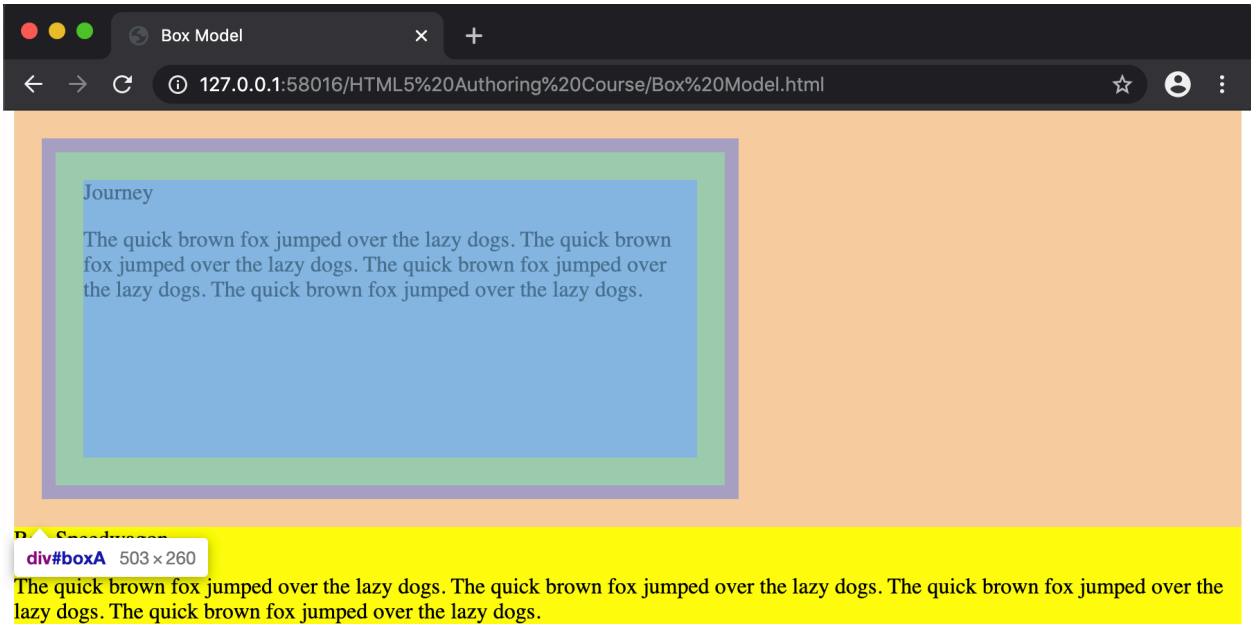
The next layer of the box model is `margin`. The margin is the space outside of the border. Margin controls the space between different elements efficiently.

Let's add a margin to `boxA`.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Box Model</title>
    <style>
      #boxA {
        background-color: lightblue;
        width: 50%;
        height: 200px;
        padding: 20px;
        border: 10px solid blue;
        margin: 20px;

      }
      #boxB {
        background-color: yellow;
      }

    </style>
  </head>
  <body>
    <div id="boxA"> Journey
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
    <div id="boxB"> Reo Speedwagon
    <p>The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.
```

```
    The quick brown fox jumped over the lazy dogs.
    The quick brown fox jumped over the lazy dogs.</p>
    </div>
  </body>
</html>
```

Here's how it manifests:
Notice that the elements move away from the page's



sides. Again, it is also possible to define different values for different sides of the box. You would do this the same way as you would define padding. You can use `margin-top`, `margin-left`, `margin-bottom`, and `margin-right` or the shorthand pattern `margin: top right bottom left;`.

Do this

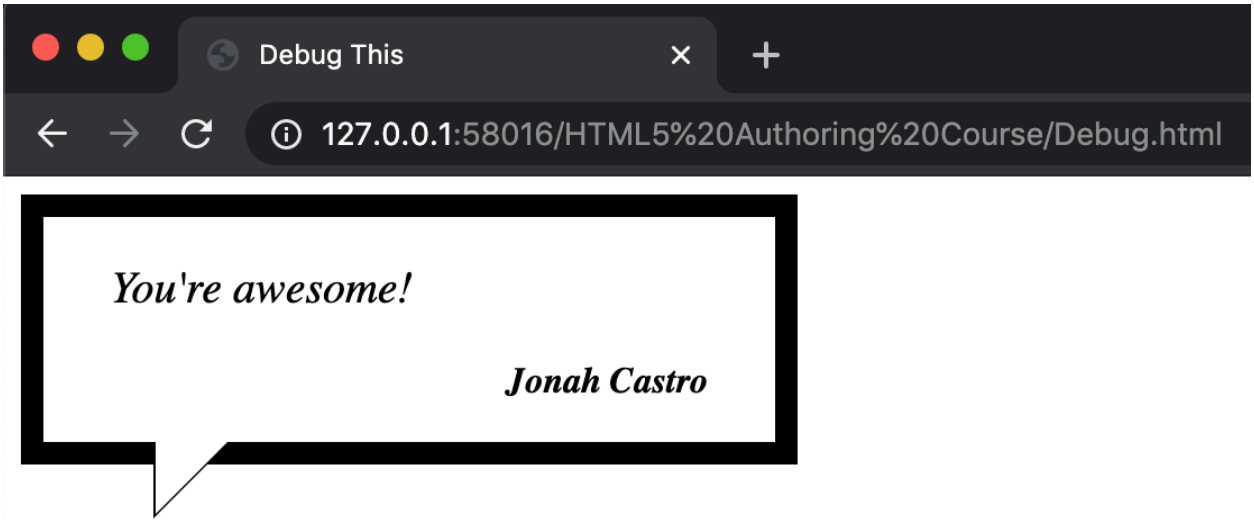
Now, it is time for you to practice on your own. Start a new HTML document and save it as `boxes.html`. Using the code above as a guide, create a document that shows at least 2 of your favorite pets (or other

animals) inside the box model. The box model contains the name, the picture, and the description of the pet. Add padding, margin, and border to make them more attractive.

Make sure to test it in the browser, so you know that it is working correctly before moving on.

Debug This:

There are errors in this code preventing it from displaying the box model correctly. Fix the errors, so the box displays correctly in your browser like this:



Here is the code to debug:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Debug This</title>
    <style>
      #testimonial{
        margin: 0 0 0 30px;
```

```
        width: 100%;
        height: 100px;
        font-style: italic;
        border: 10px dotted;
        position: relative;
      }
      #testimonial:before, #testimonial:after{
        position: absolute;
        content:"";
        width: 0;
        height:0;
        top: 100%;
        left: 15%;
        border-left: 0px solid transparent;
      }
      #testimonial:before{
        border-right: 34px solid transparent;
        border-top: 34px solid black;
        z-index: 1;
      }
      #testimonial:after{
        margin-top: -1px;
        margin-left: 1px;
        border-right: 33px solid transparent;
        border-top: 33px solid white;
        z-index: 2;
      }
      h4 {
        text-align: right;
      }
      p {
       font-size: 20px;
      }
    </style>
</head>
<body>
    <div id="testimonial">
      <p> You're awesome!</p>
      <h4> Jonah Castro </h4>
```

```
    </div>
  </body>
</html>
```

Submit This

Create an HTML5 document from scratch that is correctly formed and coded that displays a `Card Design` using box model with border, margin, and padding. The card content should have your image, name, and position.

Enjoy coding!

Remember, when submitting the work please use the following naming convention for your file: `HTMLAUTHORING_LastName_SectionName.html` . So if your last name is Smith and you are submitting this section. Your file name should be `HTMLAUTHORING_Smith_BoxModel.html` . Since you have two files for this exercise, please zip them together before uploading.

For this course go to https://www.dropbox.com/request/RhW9kBDXtisq2Fsvg3hY to submit your assignments.