Lesson 3 Challenge - Monsters

Margins, Backgrounds, and Floats

In this tutorial, you'll explore elements of the CSS box model, adjust the spacing around objects on a page, add colourful borders to items on a page, and control the size and flow of page elements.

Controlling Page Margins and Backgrounds

You'll start with a very basic HTML file containing an internal style sheet with a basic CSS reset style.

Open <u>monsters.html</u> **in Sublime (find in this lessons folder)**

There's already an internal style sheet added to this (a css reset)The styles basically remove all margins, padding, and font size from the most common block-level elements and eliminate many of the cross-browser display problems you'll encounter related to these properties.

At a bare minimum, you should always include this set of styles in every style sheet you create. Probably the most important properties are the margin and padding settings in the first style. There's enough cross-browser weirdness related to those two properties that you should always zero these out and start fresh. You'll start with something simple: a background color.

● **In the internal style sheet, click directly after the CSS comment /* end reset styles */ and add a tag selector style:**

```
html {
        background-color: #272727;
        color: #ffffff;
}
```

● **Add another style to the internal style sheet:**

```
body {
    background-color: #272727;
```

```
    font-family: "Arial Narrow", Geneva, Arial, Helvetica, sans-serif;
    width: 50%;
}
```

This style adds a dark chocolate background color to the <body> tag

Next you'll give the body tag a width and adjust its padding margins and font weight..

● **Edit the body style you just created by adding five new properties (changes are in bold)**

```
body {
    background-color: #272727;
    font-family: "Arial Narrow", Geneva, Arial, Helvetica, sans-serif;
    width: 50%;
    margin-top: 20px;
    margin-left: auto;
    margin-right: auto;
    padding: 15px;
    font-weight: lighter;
}
```

The width property constrains the body so that it's just 50% wide:

The margin-top property adds 20 pixels of space from the browser window's top edge—nudging the body tag down just a bit—while the left and right margin settings center the body in the middle of the browser window. "auto" is just another way of telling a browser, "You figure it out," and since that auto value is applied to both the left and right margins, a browser simply provides equal space on the left and right side.

● **Add a main image to the page.**

Immediately below the opening body tag add the following":

```html
<div>
    <img src="images/monsters.jpg"    alt="Monstersaurus">
</div>.
```

This is our main image / logo and will dominate the page "above the fold"

● **Save the file and preview the page in a web browser.**

The text needs some attention. You'll take care of that next.

● **Adjusting the Space Around Tags Lab**

Since the CSS reset styles pretty much stripped the text on this page of all formatting, you'll need to create styles to make the headings and paragraphs look great. You'll start with the <h1> tag at the top of the page.

Add the following style:

```css
h1 {
    font-size: 2.75em;
    font-family: "Amatic SC", cursive;
    font-weight: bolder;
    text-align: center;
    letter-spacing: 1px;
    color: rgb(249,107,24);
    text-transform: uppercase;
    background-color: #272727;
    padding: 5px 0 20px 0;
}
```

The top headline is 2.75 ems tall and all uppercase, uses Amatic SC from Google fonts, and has an orange color, with a little space between each letter. The text-align property makes sure the text is centered in the middle of the box. The padding

shorthand property provides a concise way to add padding between the image to be placed above and the content below.

● **Now, you'll add some formatting of the <h2> tags.**

Add the following style after the h1 tag style:

```css
h2 {
    font-size: 1.5em;
    font-family: "Amatic SC", cursive;
    color: rgb(249,107,24);
    padding-top: 5px;
    padding-bottom: 5px;
    margin: 15px 0 5px 0;
    overflow: hidden;
}
```

This style adds some basic text formatting. To add a bit of space between the headline text and the lines, it puts a small bit of padding at the top and bottom. The margin property
adds 15+ pixels above the headline and 5 pixels below it.
Setting the overflow property to hidden hides the borders that pass beyond the headline text and under the floating element.

● **Save the file and preview the page in a web browser.**

Building a Sidebar Lab

Sidebars are common elements in most types of print publications like magazines, books, and newspapers. They compartmentalize and highlight small chunks of information like a resource list, contact information, or a related anecdote. But to be effective, sidebars shouldn't interrupt the flow of the main story. They should, like the name says, sit unobtrusively off to one side, which you can easily make happen with CSS.

First, you must isolate the region of the page that makes up the sidebar. The <div> tag is the perfect tool. You can enclose any amount of HTML into its own self-contained chunk by wrapping it in a <div> tag.

● **Scroll down the page into the HTML and locate each div with** `class="sidebar"` and modify each div in order of appearance to include a background image as follows;

```
<div class="sidebar" style="background: url(images/29_hi.jpg) no-repeat;">
....
<div class="sidebar" style="background: url(images/21_funnzy.jpg) no-repeat;">
....
<div class="sidebar" style="background: url(images/15_neon_monster.jpg)
no-repeat;">
....
```

. Next, you'll create a style for each sidebar

⬤ **Scroll back up to the page's style sheet, and add the following style below the h2 style you created earlier:**

```
. .sidebar {
    color: #ffffff;
    width: 30%;
    float: right;
    margin: 10px;
    background-color: #6FC49C;
    padding: 10px 20px;
    border-top: 20px solid #33291C;
    border-bottom: 20px solid #33291C;
    border-radius: 10px;
    box-shadow: 5px 5px 10px rgba(0,0,0,.9);
}
```

This style sets the width of the content area (where the text appears) to 30 percent. You don't have to use an absolute value like pixels for widths. In this case, the sidebar's width is 30 percent of the width of the container. The float property moves the sidebar to the right side of the box, and the margin property adds 10 pixels of space around the sidebar. There's a top and bottom border visible with a radius of 10px. The sidebar also has a box shadow to raise it on the page

● **After the .sidebar style, in the internal style sheet, add a descendent selector:**

```
.sidebar h2 {
    color: #272727;
    font-family: 'Lobster', cursive;
    border: none;
```

```
    margin-top: 0;

    padding: 0;

}
```

Because of the .sidebar, this style is more powerful—that is, it has greater specificity—than the basic h2 style.

● **Save the file and preview the web page in a browser.**