

WORKSHOP 9.1 — Adding a little padding

In this exercise, we'll begin adding box properties to improve the appearance of the home page of the website for the fictional **Black Goose Bakery**. I've given you a head start by marking up the source (*bakery.html*). The bakery page uses an external style sheet, *bakery-styles.css*. Everything we will be doing to format this page over the next few chapters happens in the CSS file, so you should never need to edit the HTML document; however, that is the file you will open in the browser to see the results of your style changes.

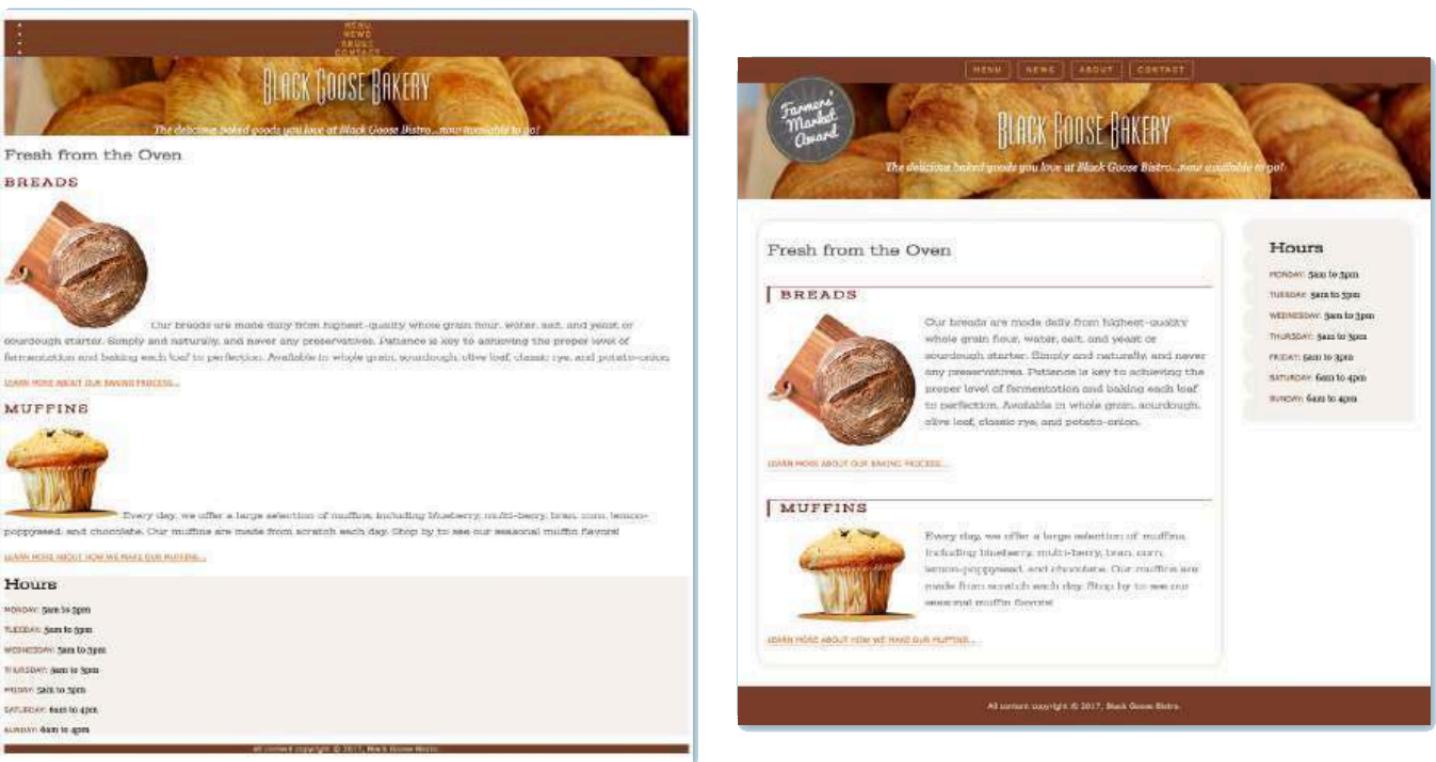


FIGURE 15.6 shows before and after shots of the home page. It's going to take several exercises to get there, and padding is just the beginning.

Start by getting familiar with the source document. Open *bakery.html* in a browser and a text editor to see what you've got to work with. The style sheet has been added with an `@import` rule in the style element. The document has been marked up with `<header>` (including a `<nav>` section), `<main>`, `<aside>`, and `<footer>` sections.

Now take a look at `bakery-styles.css` in your text editor. I used comments in the style sheet to organize the styles related to each section (bonus points for you if you keep the styles organized as you go along). You will find styles for text formatting, colors, and backgrounds—all properties that we've covered so far, so they should look familiar. Let's get started by adding some rules to `bakery-styles.css` to add padding to the elements.

NOTE This design uses a Google web font called *Stint*. You will need to have an internet connection in order to see it. If you are working offline, you will see Georgia or some serif font instead, which is just fine for these purposes, but your page won't look exactly like the ones in the figures.

Thinking Inside the Box — Padding

The exercise continues with explicit numbered steps that show which CSS rules to add. Below is the set of instructions you should add to `bakery-styles.css`.

1. The first thing we'll do is set the *box-sizing* model to `border-box` for all the elements in the document. Add these new rules to the `bakery-styles.css`. This will make measurements simpler going forward

```
html {  
  box-sizing: border-box;  
}  
  
* {  
  box-sizing: inherit;  
}
```

2. Now find the styles for the `header` and give it a height. It will fill 100% of the width of the page by default, so the width is taken care of. I picked `240px` for the height because it seemed tall enough to accommodate the content and show a nice amount of the croissant background image, but you can play around with it:

```
header {  
  ...  
  height: 240px;  
}
```

3. The `main` section is going to need a little padding, so add `16px` of padding on all sides. You can add this declaration after the existing main styles:

```
main {  
  ...  
  padding: 16px;  
}
```

4. Next, we'll get a little fancier with the `aside` element ("Hours"). We'll need extra padding on the left side for the tiling scallop background image to be visible. There are several approaches to applying different padding amounts to each side, but we're going to do it in a way that gives you experience deliberately overriding earlier declarations.

Use the `padding` shorthand property to add `16px` of padding on all sides of the `aside` element. Then write a second declaration that adds `45px` of padding on just the left side. Because the `padding-left` declaration comes second, it will override the `16px` left side setting that was applied with the shorthand property:

```
aside {  
  ...  
  padding: 16px;  
  padding-left: 45px;  
}
```

5. Finally, that footer is looking skinny and cramped. Let's add some padding to that too, which will increase the height of the footer and give the content some space:

```
footer {  
  ...  
  padding: 12px;  
}
```

6. Save the `bakery-styles.css` document, and then open (or reload) `bakery.html` in a browser to see the result of your work. The changes at this point are pretty subtle. FIGURE 15.7 uses colors to highlight the padding additions, but you will see only the added space.

FIGURE 15.7

The shaded areas indicate the padding added to `main` (blue), `aside` (pink), and `footer` (yellow). The colors were added to the figure for demo purposes but won't render in the browser.



WORKSHOP 9.2 — Border tricks

In this exercise, we'll have some fun with borders on the **Black Goose Bakery** page. In addition to putting borders around content sections of the page, we'll use borders to beef up the headlines and as an alternative to underlines under links.

Steps

1. Open `bakery-styles.css` in a text editor if it isn't already. We'll start with the basics by using the shorthand `border` property to add a tan double rule around the `main` element. Add the new declaration to the existing rule for `main` :

```
main {  
  ...  
  padding: 16px;  
  border: double 6px #eaddc4;  
}
```

2. Now try out some `border-radius` properties to add generous rounded corners to the `main` and `aside` sections. A 25-pixel radius should do. Pixels are my choice over `ems` here because I don't want the radius to scale with the text. Start by adding this declaration to the styles for `main` :

```
border-radius: 25px;
```

And give just the top-right corner of the `aside` element a matching rounded corner:

```
aside {  
  ...  
  border-top-right-radius: 25px;  
}
```

3. Just for fun (and practice), we'll add a decorative border on two sides of the baked goods headings (`h3`). Find the existing rule for `h3` elements and add a declaration that adds a 1-pixel solid rule on the top of the headline. Add another that adds a thicker 3-pixel solid rule on the left. I want the borders to be the same color as the text, so I don't need to specify the `border-color`

Finally, to prevent the text from bumping into the left border, add a little bit of padding (1em) to the left of the headline content:

```
h3 {  
  ...  
  border-top: 1px solid;  
  border-left: 3px solid;  
  padding-left: 1em;  
}
```

4. The last thing we'll do is replace the standard underline with a decorative bottom border under links. Start by turning off the underline for all links. Add this rule in the "link styles" section of the style sheet:

```
a {  
  text-decoration: none;  
}
```

Then, add a 1-pixel dotted border to the bottom edge of links:

```
a {  
  text-decoration: none;  
  border-bottom: 1px dotted;  
}
```

As is often the case when you add a border to an element, it's a good idea to also add a little padding to give the content some breathing room:

```
a {  
  text-decoration: none;  
  border-bottom: 1px dotted;  
  padding-bottom: .2em;  
}
```

Now you can save the style sheet and reload `bakery.html` in your browser. **FIGURE 15.14** shows a detail view of how your page should be looking so far.

The screenshot displays a section of a bakery website. At the top left, there is a red rectangular button labeled "MUFFINS". Below it is a large image of a muffin. To the right of the image is a paragraph of text: "Every day, we offer a large selection of muffins, including blueberry, multi-berry, bran, corn, lemon-poppyseed, and chocolate. Our muffins are made from scratch each day. Stop by to see our seasonal muffin flavors!" Below this text is a link: "LEARN MORE ABOUT HOW WE MAKE OUR MUFFINS...". On the left side, there is a section titled "Hours" with a small icon of a person. A red circle highlights the right edge of the page, indicating where the border has been added.

FIGURE 15.14 the result of out border additions.

WORKSHOP 9.3 — Adding margin space around elements

It's time to adjust the margins around the elements on the **Black Goose Bakery** page. We'll start by adjusting the margins for the whole document, and then make tweaks to each section from top to bottom. You should have `bakery-styles.css` open in a text editor.

Steps

1. It's a common practice to set the margin for the `body` element to zero, thus clearing out the browser's default margin setting. Add this margin declaration to the body styles, and then save the file and open `bakery.html` in a browser—you'll see that the elements now extend to the very edge of the window with no space in between:

```
body {  
  ...  
  margin: 0;  
}
```

Note: When the value is 0, you don't need to provide a specific unit.

2. If you're a careful observer, you may have noticed that there is still a bit of whitespace above the colored navigation bar. That happens to be the top margin of the `ul` list pushing the whole `nav` element down from the top edge of the browser window.
Let's take care of that. Add a new style rule in the "nav styles" section of the style sheet.

```
nav ul {  
  margin: 0;  
}
```

3. Margins are good for nudging elements around in the layout. For example, I'd like to move the h1 with the logotype down a bit by adding a margin to its top edge. I played around with a few values before deciding on 36px for this new style rule (you can try out some different values too):

```
header h1 {  
    margin-top: 36px;  
}
```

I'd also like the intro paragraph in the header to be a little closer to the logotype, so let's get wacky and use a negative top margin to pull it up. Add this declaration to the existing style rule:

```
header p {  
    ...  
    margin-top: -12px;  
}
```

4. Give the `main` section a `2.5%` margin on all sides:

```
main {  
    ...  
    margin: 2.5%;  
}
```

Save the file and reload the page in the browser. Then try resizing the window, paying attention to how the space all around the main section (even the top edge) grows when the viewport gets wider. If you don't enjoy that effect, you can reset the margin to a stable px value instead, but I wanted you to get a feel for how a percentage value works. Let's also add a little extra space above the h3 headings in the main area. I've chosen `2.5em`, but you can play around with different values to see what you like best:

```
h3 {  
    ...  
    margin-top: 2.5em;  
}
```

5. Finally, let's add some margins around the `aside`. We'll use different amounts and units on each side, for kicks. Put `1em` on the top, `2.5%` on the right side, `0` on the bottom, and a `75px` margin on the left. I'm going to let you do this one yourself. Can you make all those changes with one declaration? If you want to check your work, the finished version of the Black Goose Bakery page so far is available with the exercise materials for this chapter.

Save the style sheet again, and reload the page. It should look like FIGURE 15.20. This isn't the most beautiful design, particularly if your browser window is set wide. However, if you make your browser window narrower, you'll find that it wouldn't be too bad as the small-screen version in a responsive design.

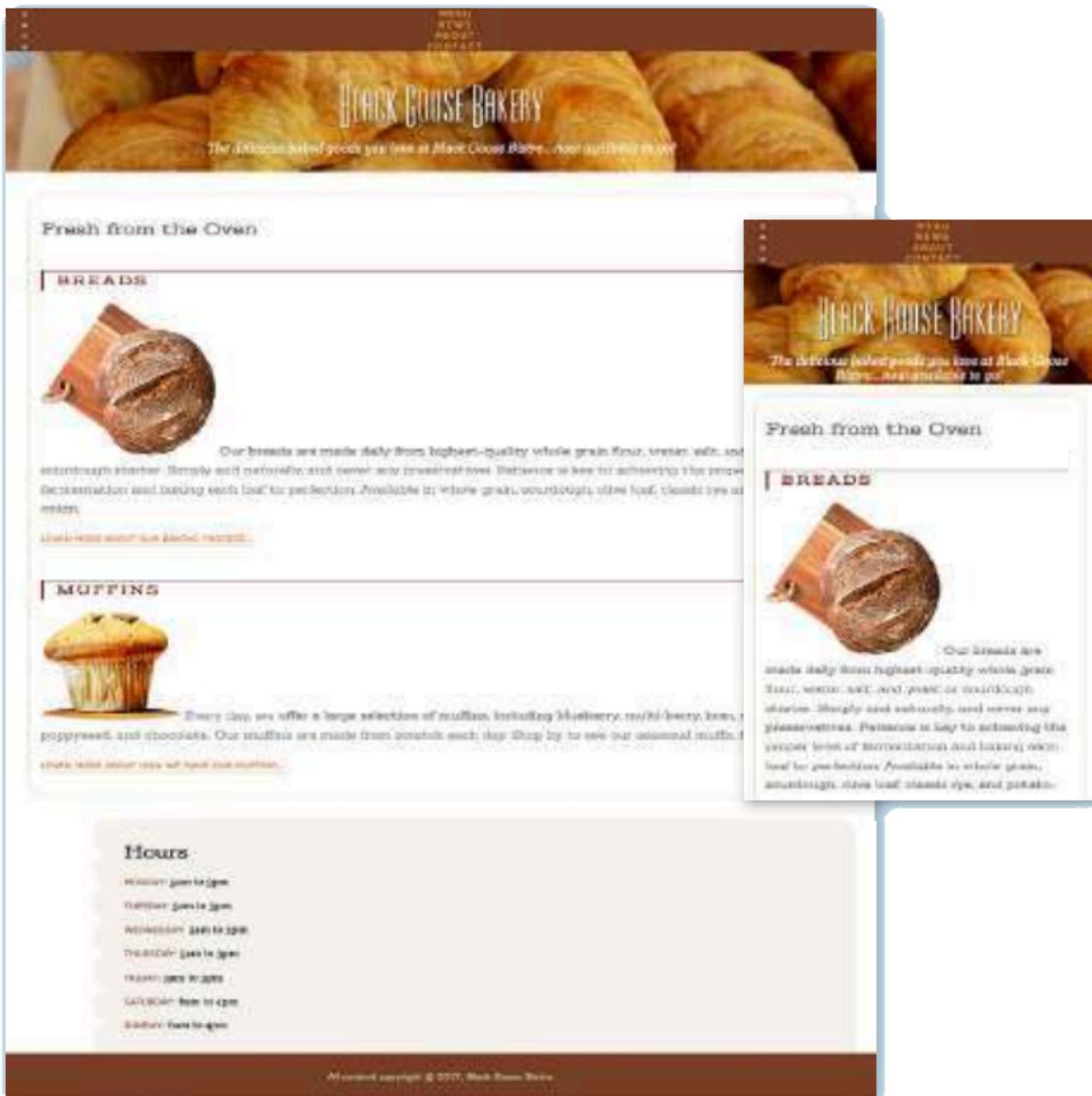


FIGURE 15.20 — The Black Goose Bakery home page after padding, borders, and margins are added.