Corrections for
*Murach’s HTML5 and CSS3 4th Edition*

These are the corrections for the significant errors in each printing of this book. In addition to the corrections listed here, you may find some trivial typos and formatting errors. All types of corrections will be made in the next printing of the book.

**How to tell which printing your book is in**

Below the copyright notation on the back of the title page (page iv), you’ll find a series of numbers like this:

10 9 8 7 6 5 4 3 2

The number on the right of this sequence tells which printing your book is. In this example, it’s the first printing (January 2019).
Corrections to the second edition

Chapter 1, pages 43 and 44

Steps 4 and 5 of exercise 1-3 are identical but should be different. Step 4 should say the following:

4. Open this file in the Chrome browser:

   `c:\murach\html5_css3_4\book_examples\ch05\12_gradients.html`

   This is the example for figure 5-12 in chapter 5. Here, the 12 in the filename refers to the figure number. As you do the exercises for this book, you may want to copy code from the examples to your exercise solution.
Corrections to the first printing

Chapter 4, page 127
The JavaScript shiv for using HTML5 semantics with IE 7 and 8 is no longer available from the location in the src attribute of the script element that’s shown at the top of this page. Instead, you can code this for the src attribute:

https://cdnjs.cloudflare.com/ajax/libs/html5shiv/3.7.3/html5shiv.js

Chapter 4, page 153
The syntax of the text-shadow property that’s shown at the top of this page indicates that you code commas between the values for this property. Instead, you code spaces as shown in the examples.

Chapter 7, page 263
The table at the top of this page should not include the SWF media form because it is obsolete.

Chapter 7, page 280
Step 18 in exercise 7-1 on this page should say to change the text to a link that plays the sampson.mp4 file, not the sampson.swf file.

Chapter 10, pages 376 to 379
These pages show how to use template areas with Grid Layout to format a web page. Although the CSS for this page works as shown, it isn’t necessary for the grid for the body of the page to contain two columns, since only one element within the body is assigned to each row. So, the CSS for the grid-template and grid-template-areas properties could be coded like this instead:

```css
grid-template: repeat(4, auto) / 1fr;
grid-template-areas
  "head"
  "navi"
  "main"
  "foot";
```

Then, when the page is reformatted for screens that are 767 pixels or less, it isn’t necessary to specify the grid-template and grid-template-areas properties for the body again because they remain the same.

Chapter 10, pages 387 and 388
The exercise on these pages has you format a web page using template areas. Like the web page in the chapter that uses template areas, this page should use a single column instead of two columns for the grid in the body. To fix this, step 3 should say to define the template so it has four rows and one column and that the column should be sized so it takes up the full width of the body. Then, in step 11, you don’t need to modify the grid for the body so it consists of a single column when the screen size is 767 pixels or less because it will already consist of a single column.

Chapter 13, page 448
The next to last paragraph indicates that the name attribute is required only for radio buttons and check boxes or when you use the get method to submit the form and you want a control name and value to be submitted to the server. However, you must code the name attribute for a control anytime you want the name and value of the control submitted to the server.
Chapter 13, pages 474 and 475

These pages indicate that the novalidate attribute can be coded on individual controls to keep them from being validated when the form is submitted. However, the novalidate attribute can be coded only on the form element to keep all of the controls on a form from being validated. Because of that, this attribute should be deleted from the Address control in the first example.

In addition to the novalidate attribute, you can code the formnovalidate attribute on a submit button to keep the controls on the form from being validated when the form is submitted using that button. This can be useful if a form contains more than one submit button.

Chapter 13, page 493

The JavaScript that’s used to set the interval for the progress and meter elements in the example on this page doesn’t display the intervals correctly. To fix that, you can delete the calls to setInterval from the setProgressAndMeter function. Then, you can code the onLoad attribute for the body element like this:

```
onLoad="setInterval(setProgressAndMeter, 500)"
```

When you do, the zero values will be displayed in both the progress and meter elements for the first one half second, and the remaining intervals will be displayed accurately.

Chapter 16, page 551

The second code example on this page includes the -webkit-transition property. This property isn’t required by any modern browsers and should be deleted.