create the web page
chapter7/starter2.html
opy the following files
pg, lighthouselogo.jpg,
lighthouse images.
he second lighthouse
We'll use the value 100

x.html. You will edit the
tion hyperlinks.

d the following styles
es.gif with no repeat.
igures the lighthouse
ue 0 in the back-
from the top (at the

use pointer passes
e selector to display
nd-position prop-
e navigation element.
igures the display at
use image appears.

Figure 7.8. Move
und images change.
er7/7.4/index.html).

If you already have a sprite graphic, check out the online tool at Sprite Cow (http://www.spritecow.com) that can generate pixel-perfect `background-position` property values for a sprite.

Checkpoint 7.1

1. Why should you organize the files in a website using folders and subfolders?
2. Which attribute configures a hyperlink to open the file in a new browser window or tab?
3. State an advantage of using CSS sprites in a website.

7.3 Three-Column CSS Page Layout

Often a web page layout will consist of a header across the top of the page with three columns below: navigation, main content, and sidebar. If you are thinking about this layout as a series of boxes—you're thinking correctly for configuring pages using CSS! Figure 7.9 shows a wireframe of this page layout design. Figure 7.10 shows a web page configured using this design. You will create this page in the next Hands-On Practice.

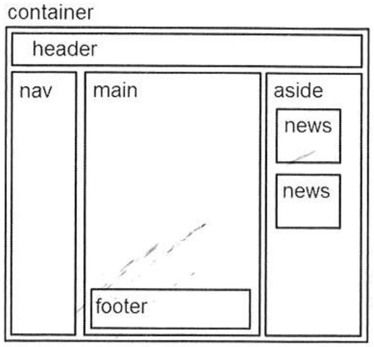
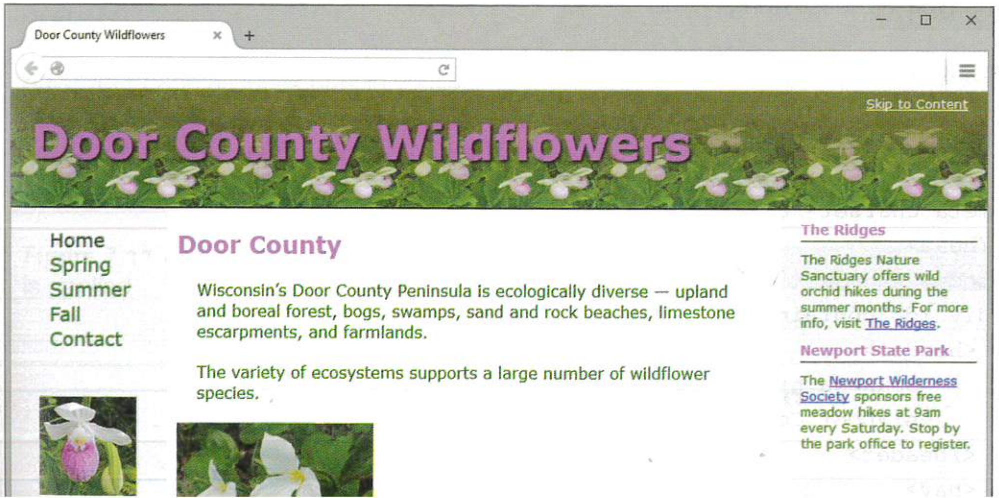


Figure 7.9 Wireframe for a three-column page layout



Door County

Wisconsin's Door County Peninsula is ecologically diverse — upland and boreal forest, bogs, swamps, sand and rock beaches, limestone escarpments, and farmlands.

The variety of ecosystems supports a large number of wildflower species.

The Ridges

The Ridges Nature Sanctuary offers wild orchid hikes during the summer months. For more info, visit The Ridges.

Newport State Park

The Newport Wilderness Society sponsors free meadow hikes at 9am every Saturday. Stop by the park office to register.

Hands-On Practice 7.5

In this Hands-On Practice, you will develop your first three-column web page using CSS. The same techniques that you used to configure the two-column page will apply here. Think of the page as a series of elements or boxes. Using a wireframe as a guide, configure the basic page structure with HTML. Then code CSS to configure page areas; use ids or classes when appropriate. Recall that a key technique for creating a two-column web page with left-column navigation is to design the left column to float to the left. A key technique for our three-column page is to code the left column with `float:left` and the right column with `float:right`. The center column occupies the middle of the browser window. Refer to Figures 7.9 and 7.10 as you complete this Hands-On Practice.

Getting Started

Locate the showybg.jpg, plsthumb.jpg, and trillium.jpg files in the chapter7/starters folder in the student files. Create a new folder called wildflowers3. Copy the files to the folder.

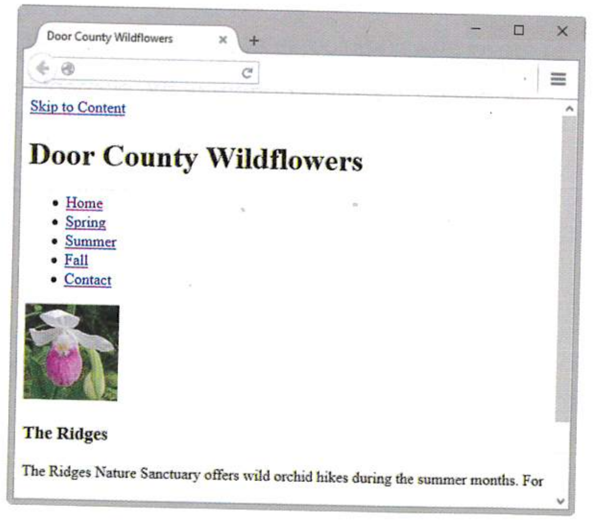
Part 1: Code the HTML

Review Figures 7.9 and 7.10. Notice the page elements: a header area with a background image; a left column with a navigation area and an image; a center column with paragraphs of text, headings, and an image that floats to the right; a right column with two news items; and a footer. You will code CSS to configure the layout and other styles. The navigation menu hyperlinks will be configured using an unordered list. As you code the HTML document, you will place the elements on the page and assign id and class values that correspond to the wireframe areas in Figure 7.9. Launch a text editor and type in the following HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Door County Wildflowers</title>
<meta charset="utf-8">
</head>
<body>
<div id="container">
  <header>
    <span><a href="#content">Skip to Content</a></span>
    <h1>Door County Wildflowers</h1>
  </header>
  <nav>
    <ul>
    <li><a href="index.html">Home</a></li>
    <li><a href="spring.html">Spring</a></li>
    <li><a href="summer.html">Summer</a></li>
    <li><a href="fall.html">Fall</a></li>
    <li><a href="contact.html">Contact</a></li>
   </ul>
   <img src="plsthumb.jpg" width="100" height="100" alt="Orchid">
   </nav>
   <aside>
```

```
    <h3>The Ridges</h3>
    <p class="news">The Ridges Nature Sanctuary offers wild orchid
hikes during the summer months. For more info, visit
<a href="http://ridgessanctuary.org">The Ridges</a>.</p>
  <h3>Newport State Park</h3>
  <p class="news">The <a href="http://newportwildernesssociety.org">
Newport Wilderness Society</a> sponsors free meadow hikes at 9am every
Saturday. Stop by the park office to register.</p>
</aside>
<main id="content">
    <h2>Door County</h2>
    <p>Wisconsin &#39;s Door County Peninsula is ecologically diverse
&mdash; upland and boreal forest, bogs, swamps, sand and rock beaches,
limestone escarpments, and farmlands.</p>
  <p>The variety of ecosystems supports a large number of wildflower
species.</p>
  <img src="trillium.jpg" width="200" height="150" alt="Trillium"
class="floatright">
    <h3>Explore the beauty <br>of Door County Wildflowers....</h3>
  <p>With five state parks, tons of county parks, and private nature
sanctuaries, Door County is teeming with natural areas for you to
stalk your favorite wildflowers.</p>
    <footer> Copyright &copy; 2018 Door County Wild Flowers<br>
  </footer>
  </main>
</div>
</body>
</html>
```

Save your page as index.html in your wildflowers3 folder. Test the page in a browser. Your display will not look like Figure 7.10 because you have not yet configured the CSS. The top of your page should look similar to the page shown in Figure 7.11.



Skip to Content

Door County Wildflowers

- Home
- Spring
- Summer
- Fall
- Contact

The Ridges

The Ridges Nature Sanctuary offers wild orchid hikes during the summer months. For

Figure 7.11 The page before CSS is applied

Part 2: Code the Basic CSS

For ease of editing, in this Hands-On Practice, you will code the CSS as embedded styles in the header section of the web page. However, if you were creating an entire website, you would most likely use an external style sheet. Launch a text editor and open index.html. Let's take a moment to consider the areas on the page shown in Figure 7.10: header, left-column navigation area, right-column sidebar area, center column, and footer. The left column will contain a navigation area and a small image. The center column will contain paragraphs, a heading, and a right-floating image. The right column will contain a series of headings and news items. Locate these areas on the sketch in Figure 7.9. Notice also that the same font is used throughout the page and the page begins immediately at the browser margin.

With your file open in a text editor, modify the head section of your document and code a `<style>` tag. Now let's consider the CSS configuration. Type the CSS in your document as indicated below:

1. Support HTML5 elements in older browsers:

```
header, nav, main, footer, aside { display: block; }
```

2. Configure box-sizing for all HTML elements:

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

3. **Body Element Selector.** Set the margin to 0. Configure the background color to #FFFFFF.

```
body { margin:0;
       background-color: #FFFFFF; }
```

4. **Container.** Configure this area with background (#EEEEEE) and text (#006600) colors, a minimum width of 960 pixels, and a font family of Verdana, Arial, or sans-serif.

```
#container { background-color: #EEEEEE;
             color: #006600;
             min-width: 960px;
             font-family: Verdana, Arial, sans-serif; }
```

5. **Header.** Configure a background color (#636631) and a background image (position showybg.jpg at the bottom of the element to repeat horizontally). Set the height to 120 pixels, text color to #CC66CC, text alignment to right, no top padding, no bottom padding, 20 pixels left padding, and 20 pixels right padding. Configure a 2 pixel solid black border across the bottom of this area.

```
header { background-color: #636631;
         background-image: url(showybg.jpg);
         background-position: bottom;
         background-repeat: repeat-x;
         height: 120px;
         color: #CC66CC;
         text-align: right;
         padding: 0 20px;
         border-bottom: 2px solid #000000; }
```

6. **Left Column.** One of the keys to this three-column page layout is that the left navigation column is designed to float to the left of the browser window. Configure a width of 150 pixels.

```
nav { float:  left;
      width: 150px; }
```

7. **Right Column.** One of the keys to this three-column page layout is that the right sidebar column is designed to float to the right of the browser window. Configure a width of 200 pixels.

```
aside { float:  right;
        width: 200px; }
```

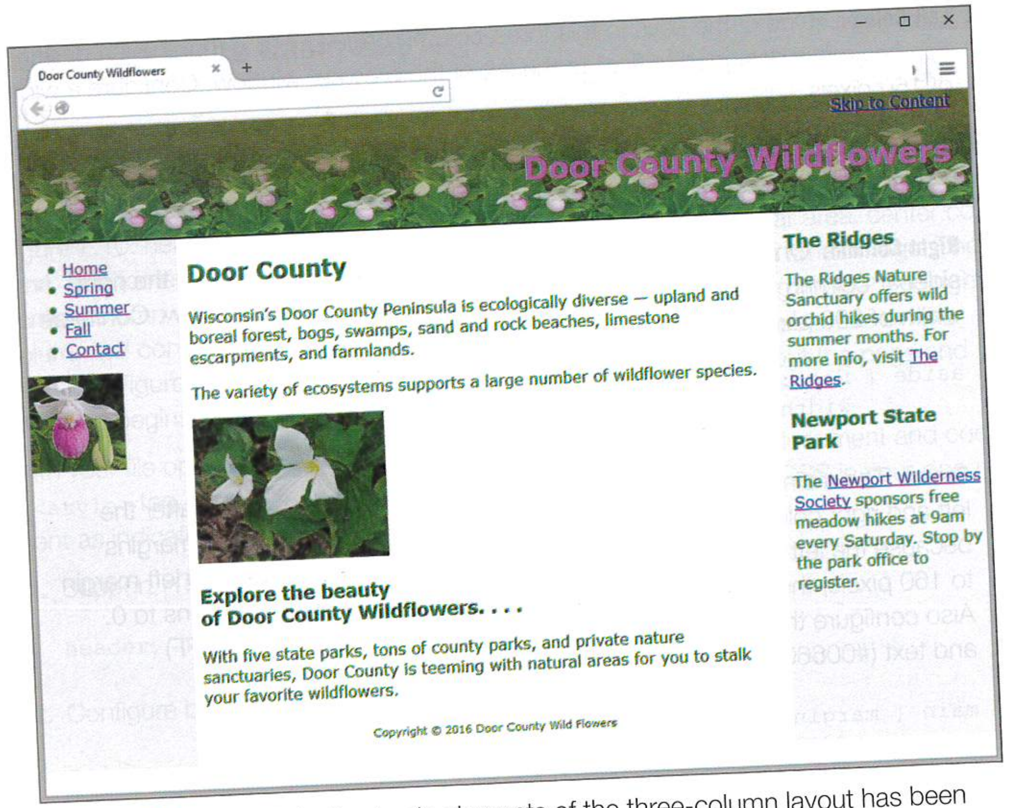
8. **Center.** The main content will take up all of the room that is available after the left and right columns float. The content area has a special need for margins because the left and right columns are floating on either side. Set the left margin to 160 pixels, the right margin to 210 pixels, and the remaining margins to 0. Also configure the padding for this area. Set the background (#FFFFFF) and text (#006600) colors.

```
main { margin: 0 210px 0 160px;
       padding: 1px 10px 20px 10px;
       background-color: #FFFFFF;
       color: #006600; }
```

9. **Footer.** Configure the page footer with very small text that is centered. Configure the background (#FFFFFF) and text (#006600) colors for this area. Set the top padding to 10 pixels. Clear the floated image in the center content area.

```
footer { font-size: .70em;
         text-align: center;
         color: #006600;
         background-color: #FFFFFF;
         padding-top: 10px;
         clear: both; }
```

At this point, you have configured the main elements of the three-column page layout. Code the closing HTML style tag with `</style>`. Save index.html in the wildflowers3 folder. It's a good idea to open your page in a browser to make sure you are on the right track. It should look similar to the one shown in Figure 7.12. Note that there is still some detail work to do, but you are well on your way!



Skip to Content

Door County Wildflowers

- Home
- Spring
- Summer
- Fall
- Contact

Door County

Wisconsin's Door County Peninsula is ecologically diverse — upland and boreal forest, bogs, swamps, sand and rock beaches, limestone escarpments, and farmlands.

The variety of ecosystems supports a large number of wildflower species.

Explore the beauty
of Door County Wildflowers. . . .

With five state parks, tons of county parks, and private nature sanctuaries, Door County is teeming with natural areas for you to stalk your favorite wildflowers.

The Ridges

The Ridges Nature Sanctuary offers wild orchid hikes during the summer months. For more info, visit The Ridges.

Newport State Park

The Newport Wilderness Society sponsors free meadow hikes at 9am every Saturday. Stop by the park office to register.

Copyright © 2016 Door County Wild Flowers

Figure 7.12 The CSS for the basic elements of the three-column layout has been completed

Part 3: Continue Coding CSS

Now you are ready to continue with your styles. Open index.html in a text editor and position your cursor on a blank line above the closing style tag.

1. **Header Area.**

   a. **The h1 Element Selector.** Notice the extra space above the "Door County Wildflowers" text, which is contained within the `<h1>` tag in the header element. You can reduce this extra space by setting a 0 top margin for the h1 element selector. Also configure left alignment, text shadow, and a font size of 3em for the h1 selector.

```
h1 { margin-top: 0;
     font-size: 3em;
     text-align: left;
     text-shadow: 2px 2px 2px #000000; }
```

   b. **Skip to Content.** Configure the "Skip to Content" hyperlink in the header area with a 0.80em font size. Also configure pseudo-classes for `:link`, `:visited`, `:hover`, `:active`, and `:focus` with text color as shown below.

```
header a {font-size: 0.80em; }
header a:link, header a:visited { color: #ffffff; }
header a:focus, header a:hover { color: #eeeeee; }
```

2. **Left Navigation Column.**

   a. **Navigation Menu.** Configure the unordered list in the `nav` element selector to provide for a 20 pixel top margin and not to display any list markers.

```
nav ul { margin-top: 20px;
         list-style-type: none; }
```

   The navigation links should have no underline (`text-decoration: none`) and a 1.2em font size. Configure pseudo-classes for `:link`, `:visited`, `:hover`, `:active`, and `:focus` with text color as shown below.

```
nav a { text-decoration: none;
        font-size: 1.2em; }
nav a:link { color:#006600;}
nav a:visited { color: #003300; }
nav a:focus, nav a:hover { color: #cc66cc; }
nav a:active { color: #000000;}
```

   b. **Left Column Image.** Configure images in the `nav` element with a margin of 30 pixels.

```
nav img { margin: 30px;}
```

3. **Main Content.**

   a. **Paragraphs.** Configure the main content's paragraph element selector to display with a margin of 20 pixels.

```
main p { margin: 20px; }
```

   b. **Headings.** Configure the h2 and h3 element selectors in the main content with the same text color as the logo header text and the same background color as the main body of the page.

```
main h2, main h3 { color: #cc66cc;
                   background-color: #ffffff; }
```

   c. **Image Floating at the Right.** Create a `floatright` class to use a 10 pixel margin and float to the right.

```
.floatright { margin: 10px;
              float: right; }
```

4. **Right Sidebar Column. This column is contained within the aside element.**

   a. **Headings.** Configure the h3 element selector in this area with a 1 pixel black solid bottom border, 2 pixels of padding at the bottom, a 10 pixel margin, 0.90em font size, and the same text color as the logo header text.

```
aside h3 { padding-bottom: 2px;
           border-bottom: 1px solid #000000;
           margin: 10px;
           font-size: 0.90em;
           color: #cc66cc; }
```

b. **News Items.** Configure a class called `news` that uses a small font and has a 10 pixel margin.

```
.news { font-size: 0.80em;
        margin: 10px; }
```

Save index.html in the wildflowers3 folder.

Part 4: Test the Page

Now that your styles are coded, test the index.html page again. Your page should look similar to the screenshot shown in Figure 7.10. If there are differences, verify the id and class values in your HTML. Also check the syntax of your CSS. You may find the W3C CSS Validation Service at http://jigsaw.w3.org/css-validator to be helpful when verifying CSS syntax. The student files contain a copy of this page in the chapter7/7.5 folder.

7.4 CSS Styling for Print

Even though the "paperless society" has been talked about for decades, the fact is that many people still love paper and you can expect your web pages to be printed. CSS offers you some control over what gets printed and how the printouts are configured. This is easy to do using external style sheets. Create one external style sheet with the configurations for browser display and a second external style sheet with the special printing configurations. Associate both of the external style sheets to the web page using two link elements. The link elements will utilize the **`media` attribute**, which indicates the **media type** for which the styles are intended, such as screen display or print display. See a list of media attribute values in Table 7.1.

Table 7.1 The link element's media attribute

Value	Purpose
`screen`	Default value; Indicates the style sheet that configures the typical browser viewport display on a color computer screen
`print`	Indicates the style sheet that configures the printed formatting
`handheld`	Although this value is intended by the W3C to indicate the style sheet that configures display on handheld mobile devices, in practice, the attribute value is not reliably supported. The next section will describe other methods for configuring the design of mobile web pages.

Modern browsers will use the correct screen or print style sheet, depending on whether they are rendering a screen display or preparing to print a document. Configure the link element for your browser display with `media="screen"`. Configure the link element for your printout with `media="print"`. An example of the HTML follows:

```
<link rel="stylesheet" href="wildflower.css" media="screen">
<link rel="stylesheet" href="wildflowerprint.css" media="print">
```

Print Styling Best Practices

You might be wondering how a print style sheet should differ from the CSS used to display the web page in a browser. Commonly used techniques for styling print are listed below.