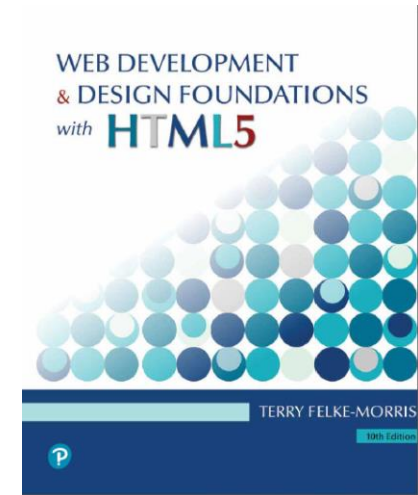# Web Development & Design Foundations with HTML5
10th Edition

CHAPTER 6
PAGE LAYOUT BASICS

Learning Outcomes

In this chapter, you will learn how to ...

◦ Describe and apply the CSS Box Model
◦ Configure width and height with CSS
◦ Configure margin with CSS
◦ Configure float with CSS
◦ Configure fixed, relative, absolute, and sticky positioning with CSS
◦ Create two-column page layouts using CSS
◦ Configure navigation in unordered lists and style with CSS
◦ Add interactivity to hyperlinks with CSS pseudo-classes
◦ Configure a hyperlink to a named fragment internal to a web page
◦ Configure images with CSS sprites
◦ Configure CSS for printing
◦ Configure a single page website with parallax scrolling

Width and Height with CSS

Table 6.1

Unit	Purpose
px	Configures a fixed number of pixels as the value
em	Configures a value relative to the font size
%	Configures a percentage value of the parent element
vh	Configures a value relative to 1% of the viewport height
vw	Configures a value relative to 1% of the viewport width

CSS Width and Height Properties

- width property
  - Configures the width of an element's content

h1 { width: 80%; }

- min-width property
  - Configures minimum width of an element
- max-width property
  - Configures the maximum width of an element

- height property
  - Configures the height of an element

h1 { height: 100px; }

The Box Model

Content
- Text & web page elements in the container

Padding
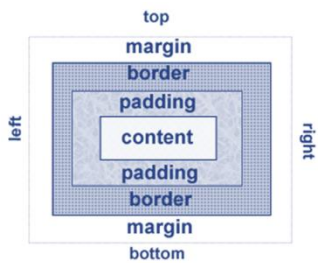- Area between the content and the border

Border
- Between the padding and the margin

Margin
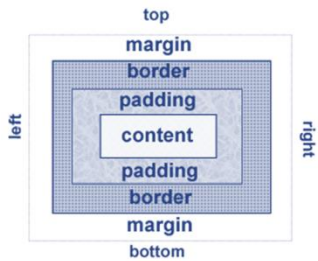- Determines the empty space between the element and adjacent elements

Figure 6.6

Configure Margin with CSS

- The margin property
- Related properties:
  - margin-top, margin-right, margin-left, margin-bottom
- Configures empty space between the element and adjacent elements

- Syntax examples

h1 { margin: 0; }

h1 { margin: 20px 10px; }

h1 { margin: 10px 30px 20px; }

h1 { margin: 20px 30px 0 30px; }

Configure Padding with CSS

- The padding property
- Related properties:
  - padding-top, padding-right, padding-left, padding-bottom
- Configures empty space between the content of the HTML element (such as text) and the border

- Syntax examples

  h1 { padding: 0; }

  h1 { padding : 20px 10px; }

  h1 { padding : 10px 30px 20px; }

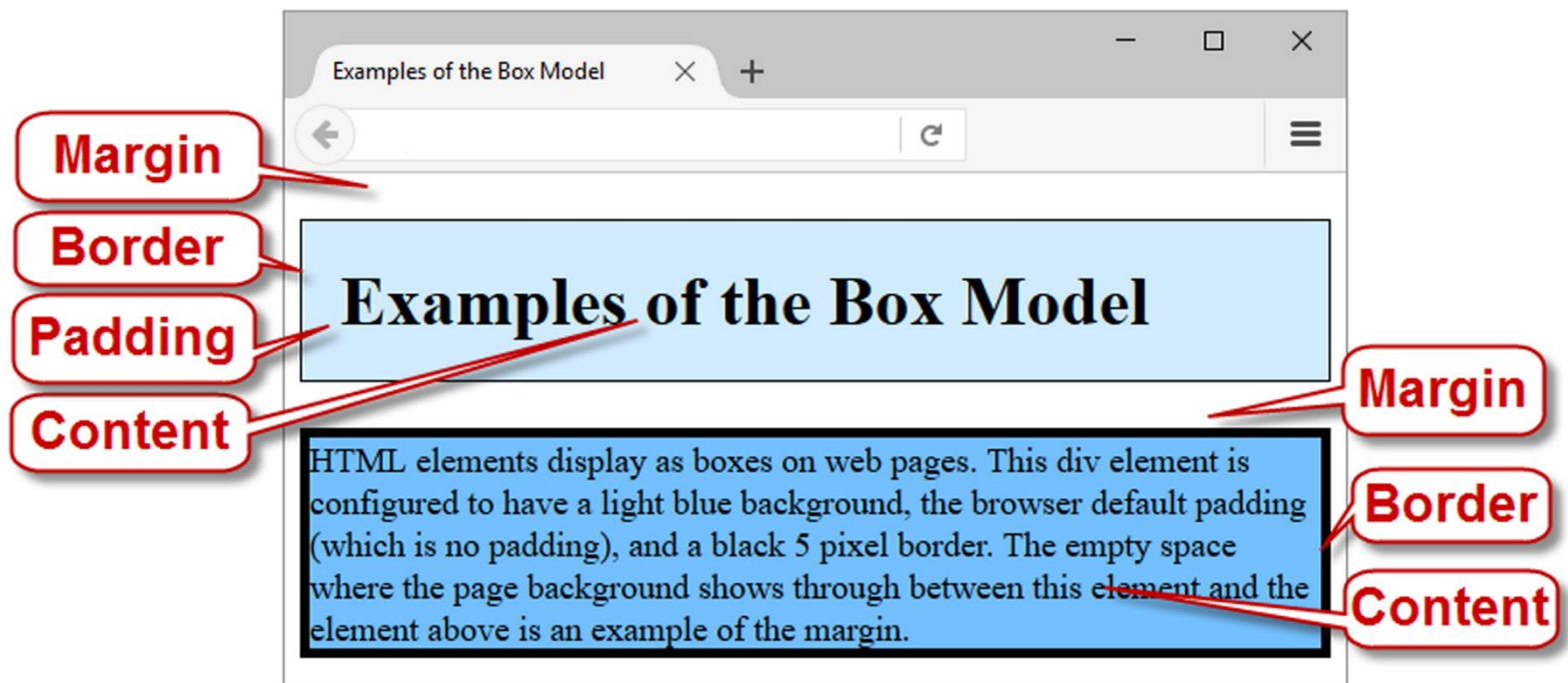 h1 { padding : 20px 30px 0 30px; }

Box Model in Action

Figure 6.7

The CSS box-sizing Property

Default value for width or height is the value for ONLY the content (not including border and padding).

The box-sizing property with border-box value
directs the browser to calculate the width and height of an element to *include* the value for content, padding, and border.

Use the universal selector (*) to apply this to all the element on the page

Example:

* { box-sizing: border-box; }

Normal Flow

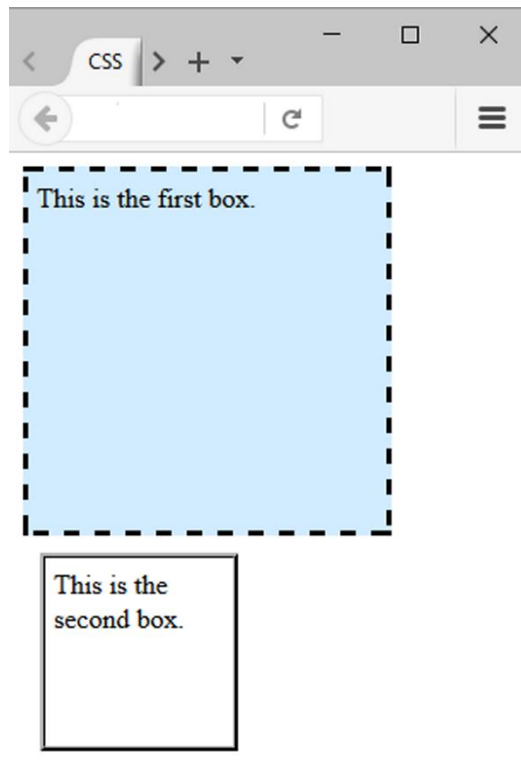Browser display of elements in the order they are coded in the web page document

This is the first box.

This is the second box.

Figure 6.8

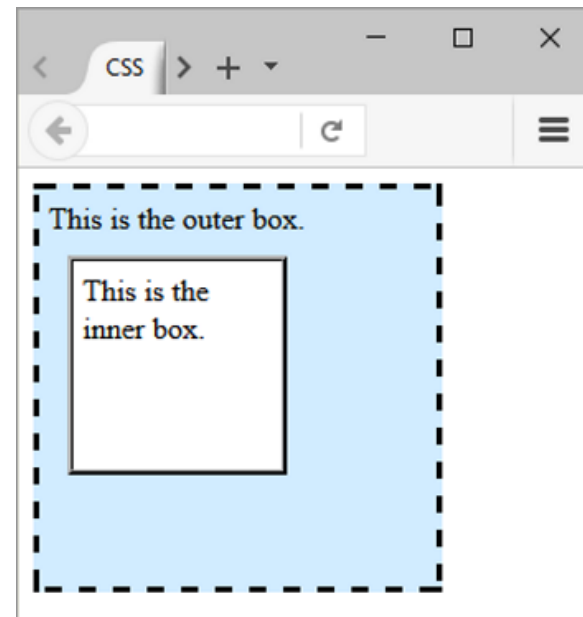This is the outer box.

This is the inner box.

Figure 6.9

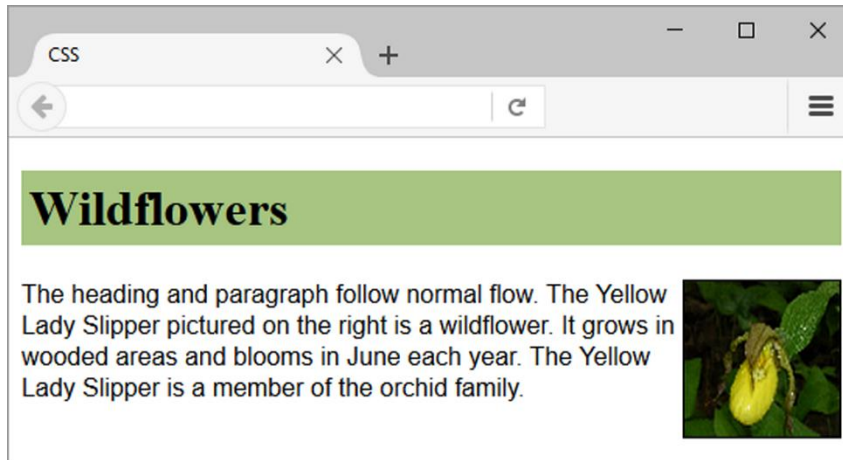# float Property

Figure 6.10

```
h1 { background-color: #CCCCCC;
     padding: 5px;
     color: #000000;
}
p { font-family: Arial,sans-serif;
}
#yls { float: right;
       margin: 0 0 5px 5px;
       border: 1px solid #000000;
}
```

Elements that seem to "float" on the right or left side of either the browser window or another element are often configured using the float property.

clear Property

The h2 text is displayed in normal flow.

Figure 6.11

Apply clear: left;
to a line break element within the div element. Now, the h2 begins on a new line under the floated image.

Useful to "clear" or terminate a float
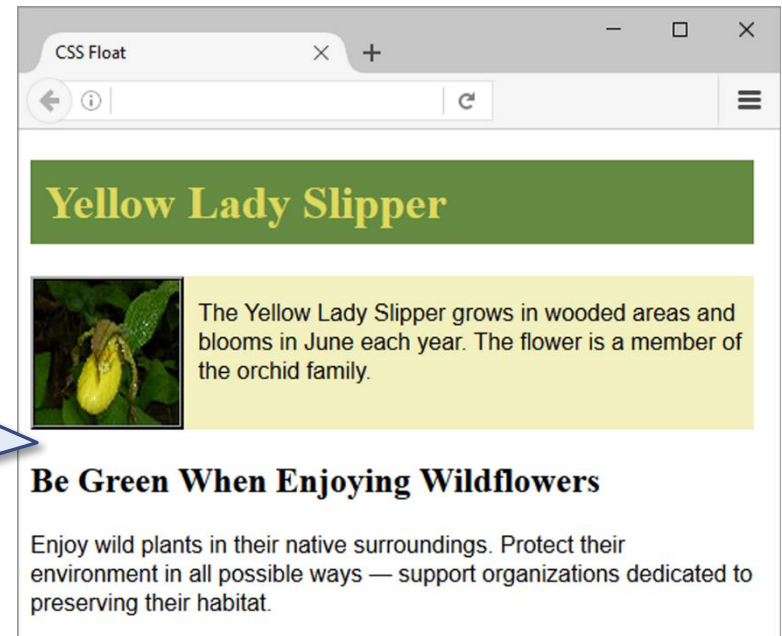
Values are left, right, and both

Figure 6.12

overflow Property

Intended to configure the display of elements on a web page.

However, it is useful to "clear" or terminate a float before the end of a container element

Values are auto, hidden, and scroll

> The background does not extend as far as you'd expect.

Figure 6.11

> overflow: auto;
> was applied to the div that contains the image and paragraph.
> Now the background extends and the h2 text displays AFTER the floated image.
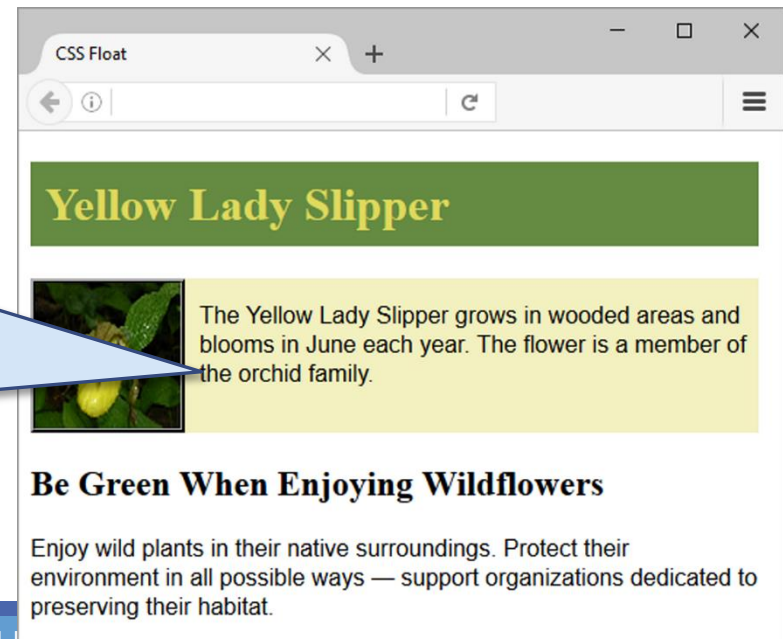
Figure 6.14

Checkpoint

1. List the components of the box model from innermost to outermost.

2. Describe the purpose of the CSS float property.

3. Which two CSS properties can be used to clear a float?

Page Layout
Single Column -> Two Column

**Single Column
Wireframe**

**Two Column
Wireframe**

Figure 6.18

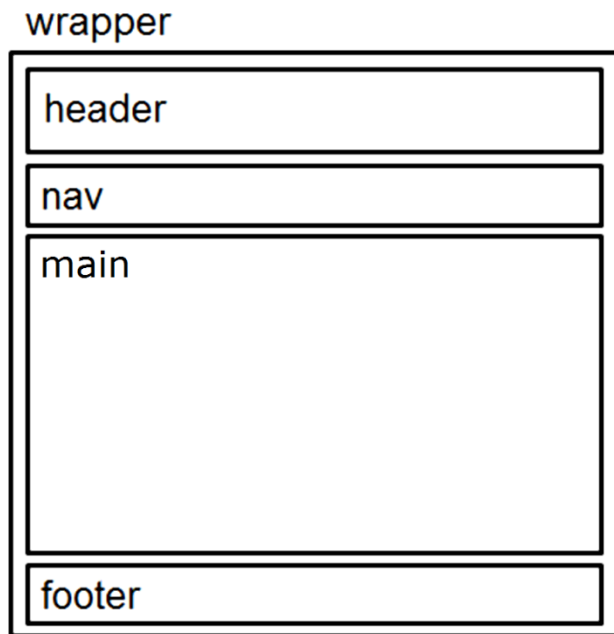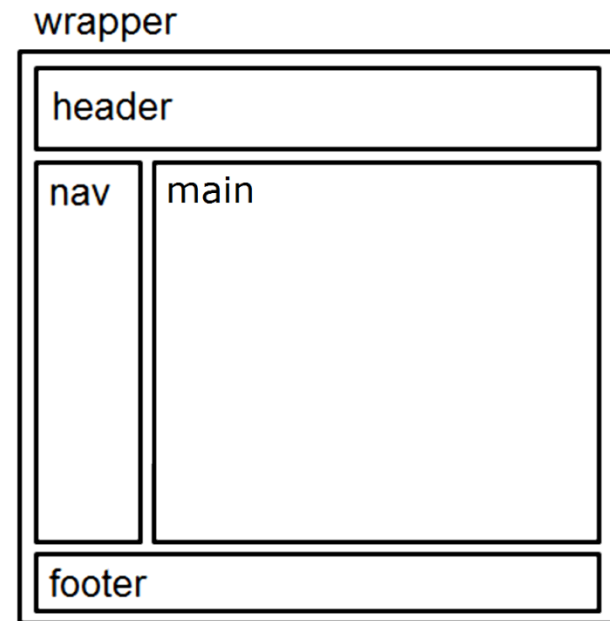Figure 6.19

Basic Two-Column Layout

Figure 6.19

```
<body>
<div id="wrapper">
  <header> <header>
  <nav> </nav>
  <main> </main>
  <footer> </footer>
</div>
</body>
```
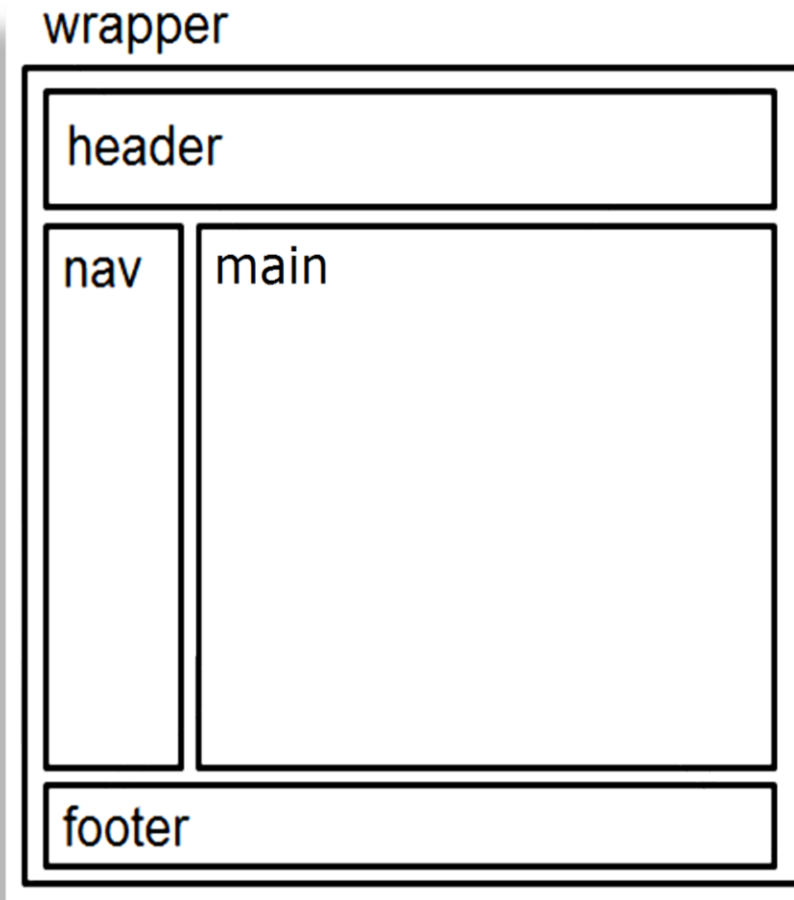
Figure 6.2

```
#wrapper { width: 80%;
        margin-left: auto;
        margin-right: auto;
        background-color: #EAEAEA; }
header { background-color: #CCCCFF; }
h1 { margin: 0; padding: 10px; }
nav { float: left;
        width: 90px;
        padding: 10px; }
main { margin-left: 100px;
        padding: 10px;
        background-color:  #FFFFFF; }
footer { text-align: center;
        font-style: italic;
        background-color: #CCCCFF; }
```
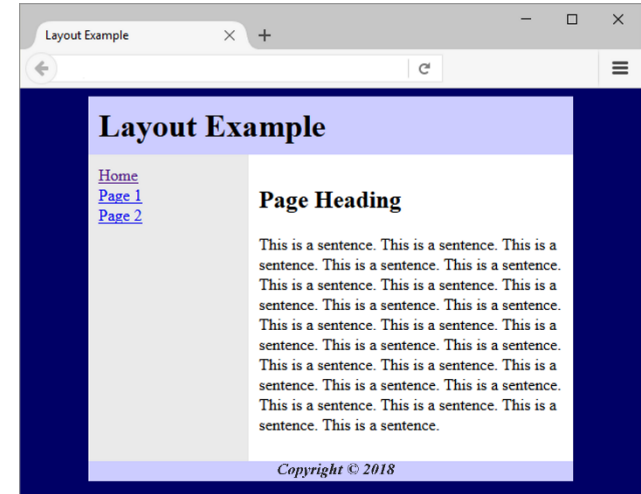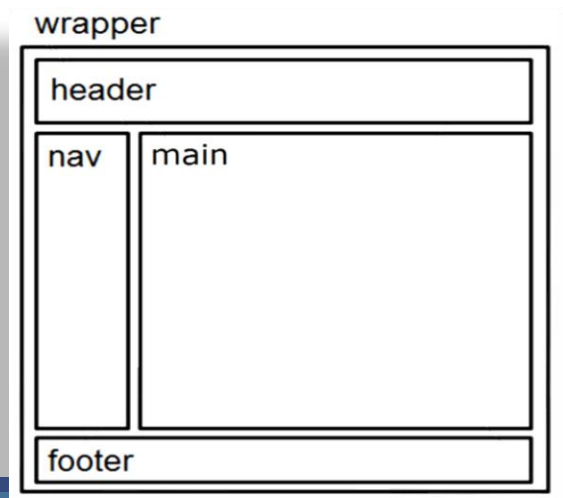

Figure 6.23

Figure 6.19

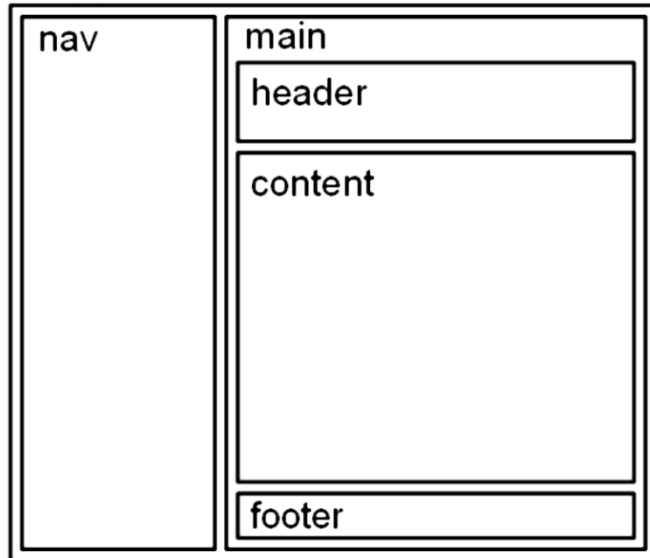# CSS Page Layout
Two Columns (left nav)
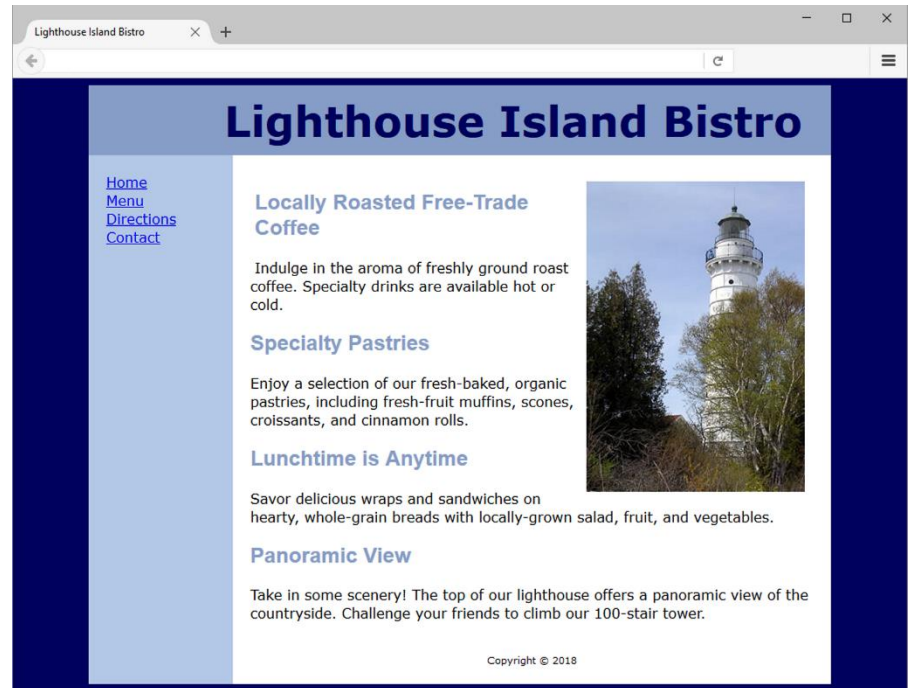

Figure 6.24

Figure 6.25

CSS display Property

Configures how and if an element is displayed

- **display: none;**
  - The element will not be displayed.

- **display: block;**
  - The element renders as a block element – even if it is actually an inline element, such as a hyperlink.

- **display: inline;**
  - The element renders as an inline element – even if it is actually a block element – such as a <li>.

- **display: inline-block;**
  - The element will display as an inline display element adjacent to other inline display elements but also can be configured with properties of block display elements including width and height.

- **display: flex;**
  - The element displays as a block-level flex container (Chapter 7)

- **display: grid;**
  - The element displays as a block-level grid container (Chapter 7)

Vertical navigation

```
<nav>

<ul>

    <li><a href="index.html">Home</a></li>

    <li><a href="menu.html">Menu</a></li>

    <li><a href="directions.html">Directions</a></li>

    <li><a href="contact.html">Contact</a></li>

</ul>

</nav>
```
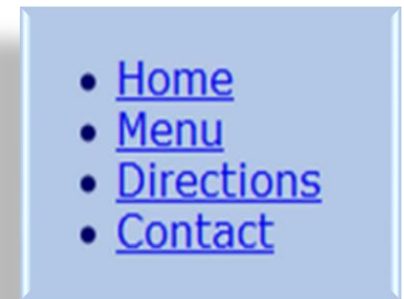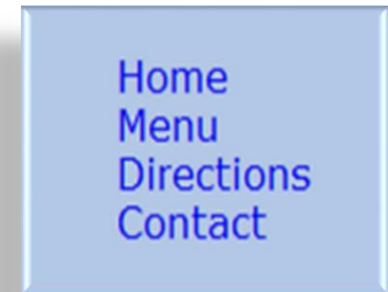

Figure 6.29

CSS removes the list marker and underline:
```
nav ul { list-style-type: none; }
nav a { text-decoration: none; }
```


Figure 6.31

Horizontal Navigation

Figure 6.32

HTML:

<nav>

    Home

    Menu

    Directions

    Contact

</nav>

CSS removes the list marker, removes the underline,
adds padding, and configures the list items for inline display.

```
nav ul { list-style-type: none;}
nav a { text-decoration: none;
        padding-right: 10px; }
nav li {  display: inline; }
```

CSS Pseudo-classes

- Pseudo-classes and the anchor element
  - **link** – default state
for a hyperlink

  - **visited** – a hyperlink that
has been visited

  - **focus** – triggered when
the hyperlink has focus

  - **hover** – triggered when
the mouse moves over the hyperlink

  - **active** – triggered when the hyperlink is being clicked

```
a:link     {color:#000066;}
a:visited {color:#003366;}
a:focus    {color:#FF0000;}
a:hover   {color:#0099CC;}
a:active  {color:#FF0000;}
```

1. Text hyperlinks are underlined by default.

Print This Page

2. The `hover` pseudo-class is triggered by the mouse. The browser no longer displays the underline below the hyperlink.

Print This Page

Figure 6.33

a:link { color: #ff0000; }

a:hover { text-decoration: none;

color: #000066; }

CSS Sprites

Sprite

◦ an image file that contains multiple small graphics
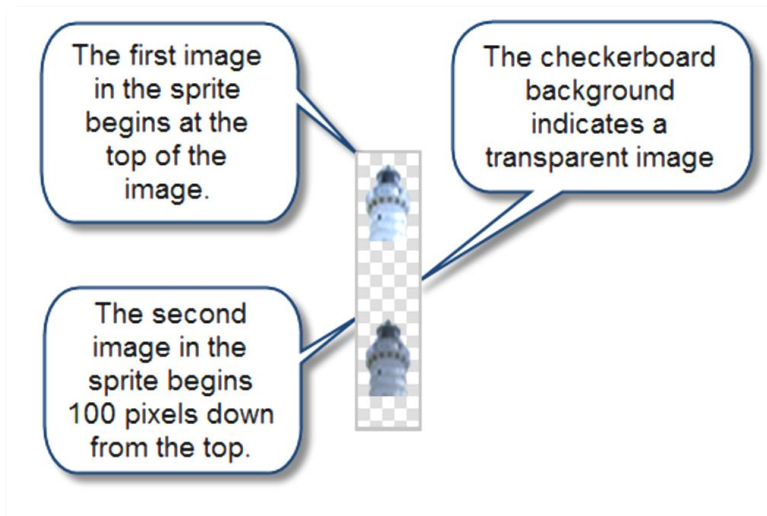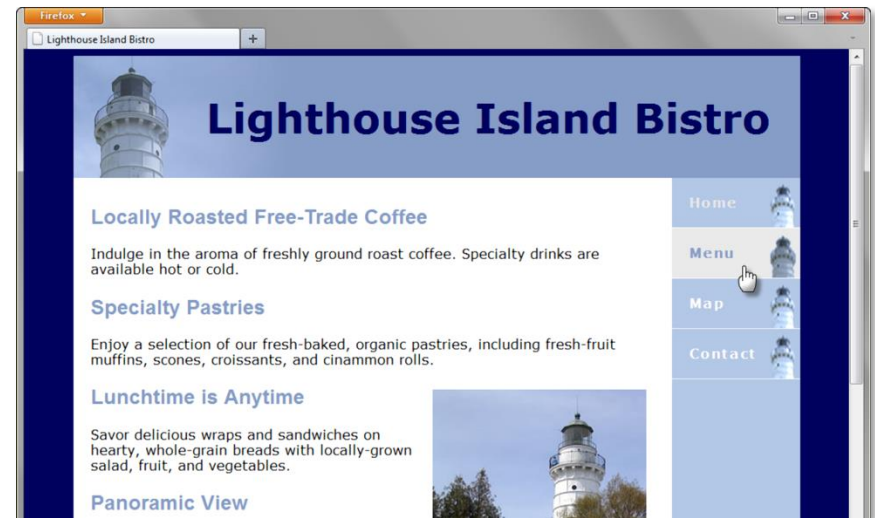
◦ advantage: saves download time

Figure 6.40

Figure 6.41

CSS Styling for Print

Create an external style sheet with the configurations for browser display.

Create a second external style sheet with the configurations for printing.

Connect both of the external style sheets to the web page using two **<link >** elements.

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

Print Styling Best Practices

Hide non-essential content
Example:

     #nav { display: none; }

Configure font size and color for printing
- Use pt font sizes, use dark text color

Control page breaks
Example:
.newpage { page-break-before: always; }

Position Property

Value	Purpose
static	Default value; the element is rendered in normal flow
fixed	Configures the location of an element within the browser viewport; the element does not move when the page is scrolled
relative	Configures the location of an element relative to where it would otherwise render in normal flow
absolute	Precisely configures the location of an element outside of normal flow
sticky	Combines features of relative and fixed positioning

Table 6.8

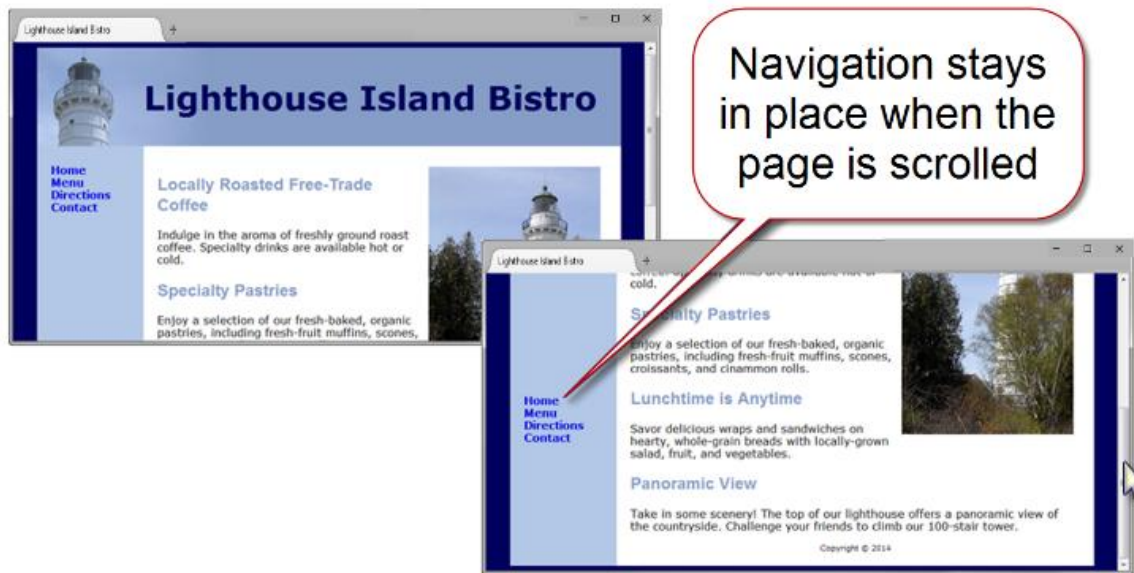# Fixed Positioning

nav { position: fixed; }

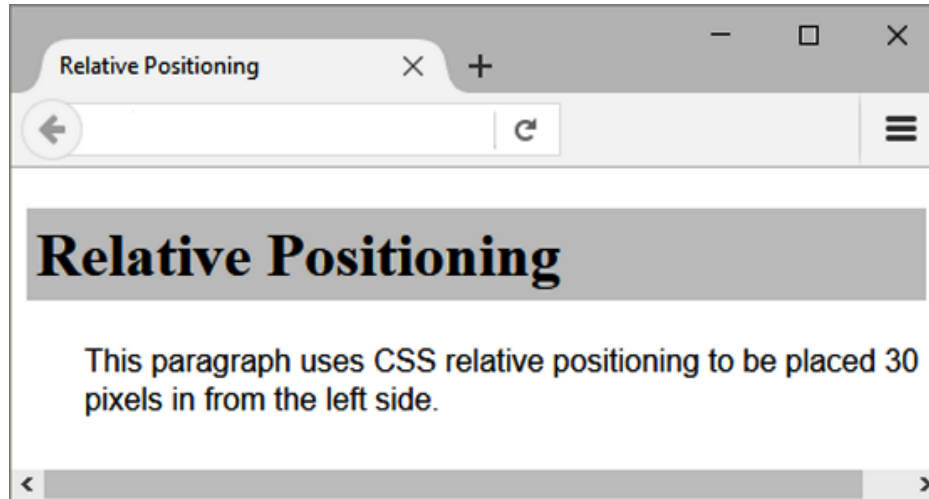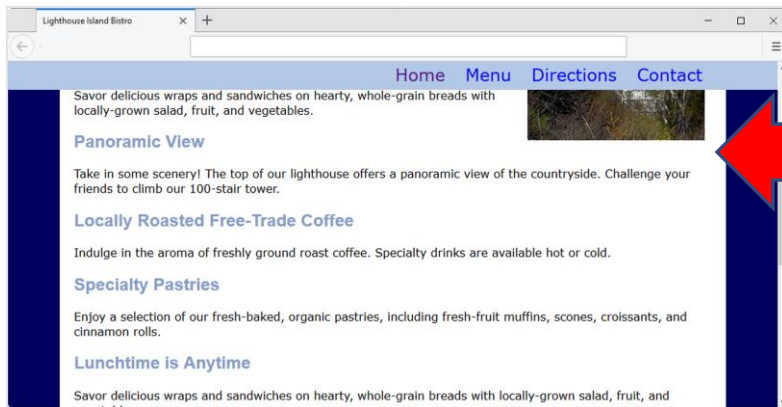Figure 6.43

Relative Positioning

Figure 6.44

```
p { position: relative;
    left: 30px;
    font-family: Arial, sans-serif; }
```

Changes the location of an element in relation to where it would otherwise appear in normal flow

Sticky Positioning

Figure 6.49

"Sticky" Navigation Bar

```
nav { position: sticky;
      top: 0;
```

- Combines features of relative and fixed positioning
- Element rendered in normal flow and then, when reached during scrolling, sticks to the specified position and remains there
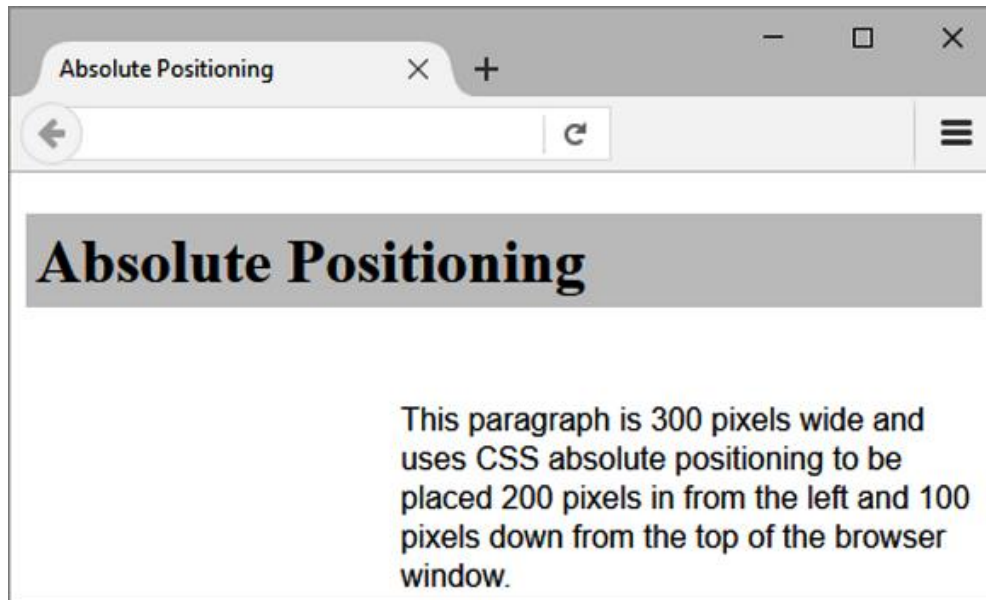
Absolute Positioning

Figure 6.48

Precisely specifies the location of an element outside of normal flow in in relation to its first parent non-static element

```
p { position: absolute;
    left: 200px;
    top: 100px;
    font-family: Arial, sans-serif;
    width: 300px; }
```

Stacking Order with the z-index Property

- Configure a third dimension: the stacking order of positioned elements

- Default z-index is 0

- Elements with higher z-index values will stack on top of elements with lower z-index values

- Example: Fixed Position Navigation Bar

  nav { position: fixed;
         top: 0; left: 0;
         height: 40px;
         width: 100%;
         min-width: 40em;
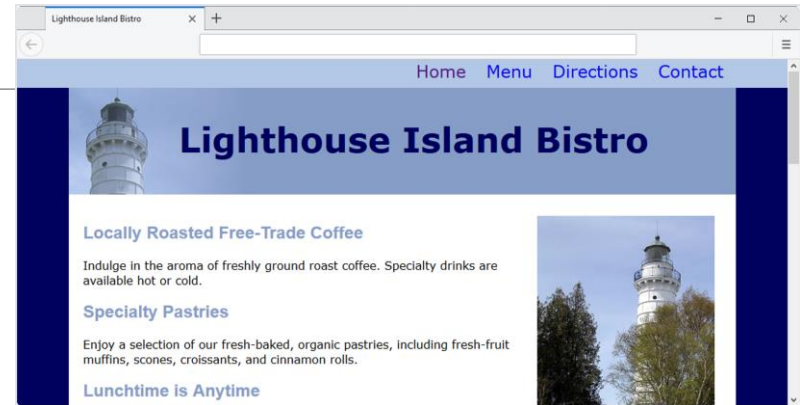   background-color: #B3C7E6; }
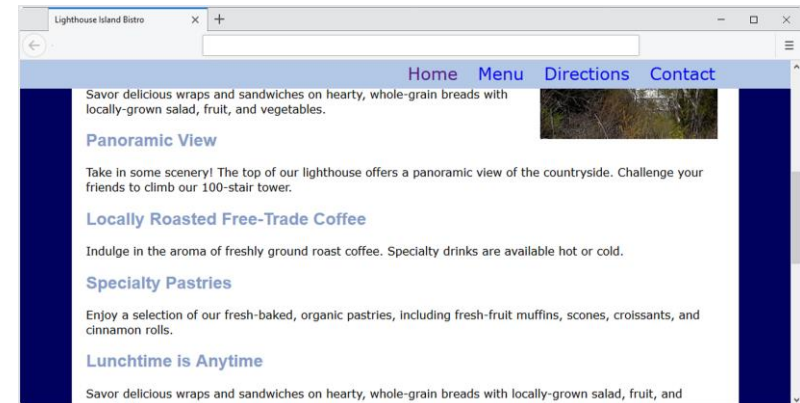

Figure 6.48

Figure 6.49

HTML Linking to Fragment Identifiers

- A hyperlink to a part of a web page

- Also called named fragments, fragment ids

- Two components:
  1. The element that identifies the named fragment of a web page. This requires the id attribute.

     <div id="top"> ….. </div>

  2. The anchor tag that links to the named fragment of a web page. This uses the href attribute.

     Back to Top

  Note the use of the # in the anchor tag!
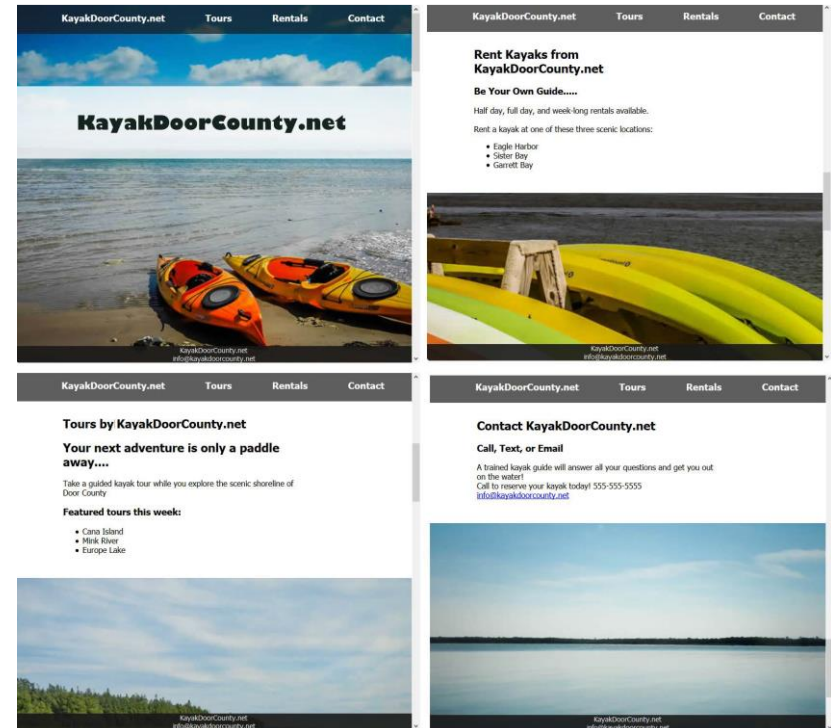
Single Page Website

Figure 6.52

Figure 6.51

- One very long page (a single HTML file)

- Clearly defined navigation area

- Navigation links to specific sections that function as a "page"

- Technique: hyperlinks to fragment identifiers

Parallax Scrolling

- Page layout technique in which the background images scrolled at a different speed than the text content

- Often accomplished with JavaScript and advanced CSS

- Most basic implementation:
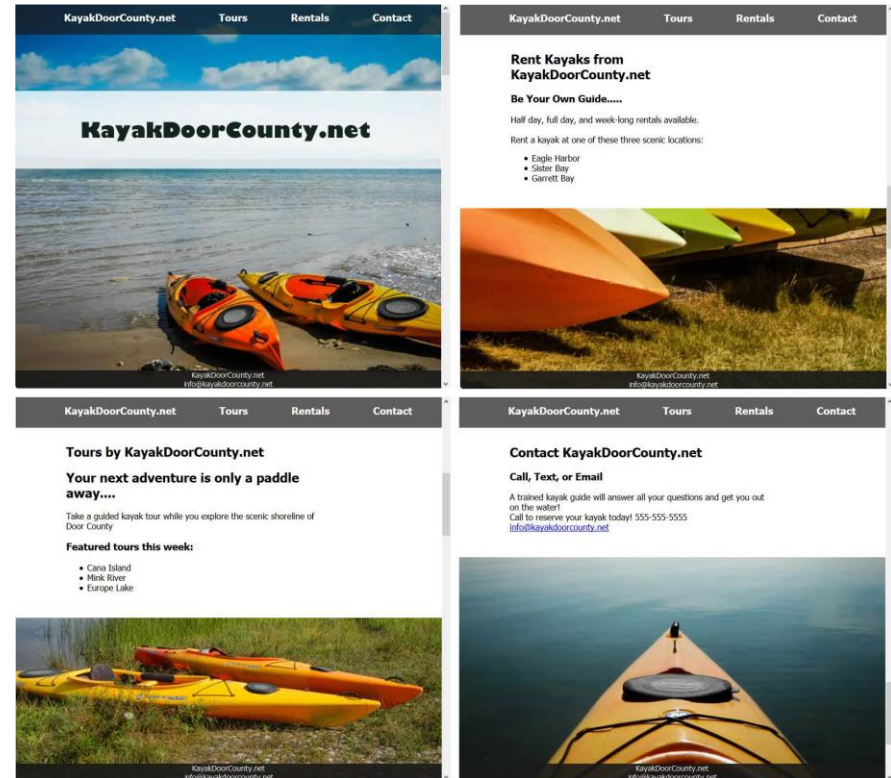 - CSS background-attachment property

  background-attachment: fixed;

Figure 6.53

CSS Debugging Tips

Manually check syntax errors

Use W3C CSS Validator to check syntax errors
- http://jigsaw.w3.org/css-validator/

Configure temporary background colors

Configure temporary borders

Use CSS comments to find the unexpected
/* the browser ignores this code */

Don't expect your pages to look exactly the same in all browsers!

Be patient!

Checkpoint

1. State an advantage of using CSS to style for print.

2. State an advantage of using CSS sprites in a website.

3. Describe a technique to keep an HTML element, such as a nav element, displayed at the top of the browser viewport even while the browser is scrolled.

Summary

This chapter introduced you to the box model, CSS pseudo-classes, configuring two-column page layouts with CSS, positioning elements, and configuring single page websites with CSS.