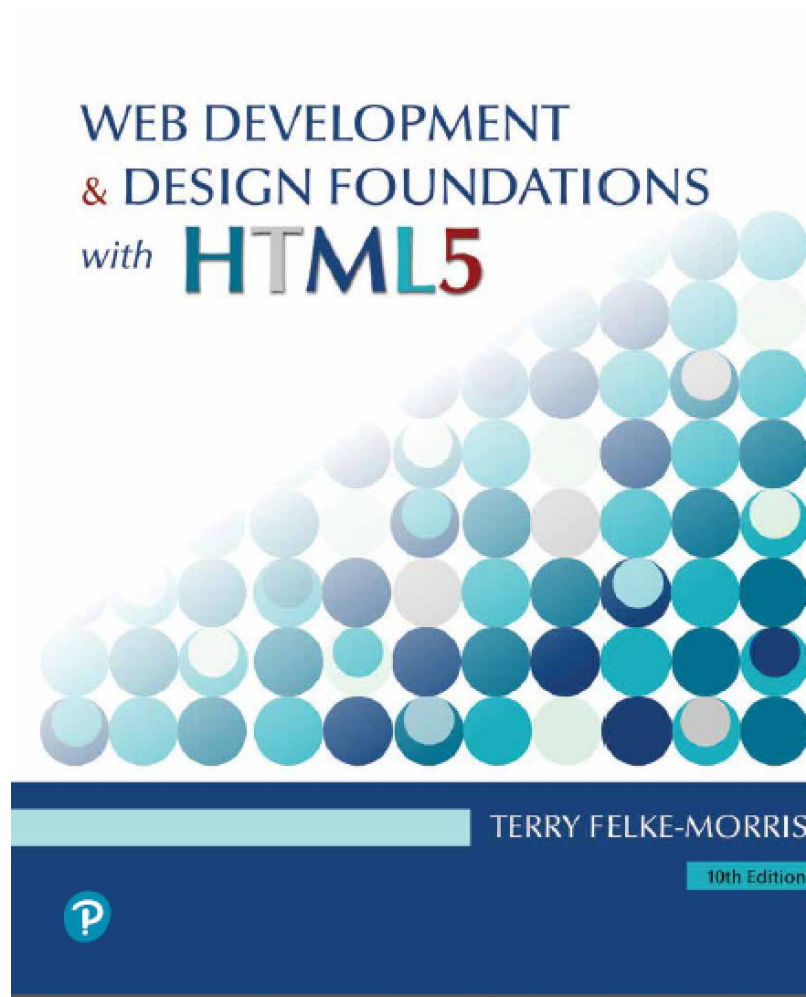


# Web Development & Design Foundations with HTML5

Tenth Edition



## Chapter 6

### Page Layout Basics

# Learning Outcomes (1 of 2)

- Configure width & height with CSS
- Describe and apply the CSS Box Model
- Configure margin and padding 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

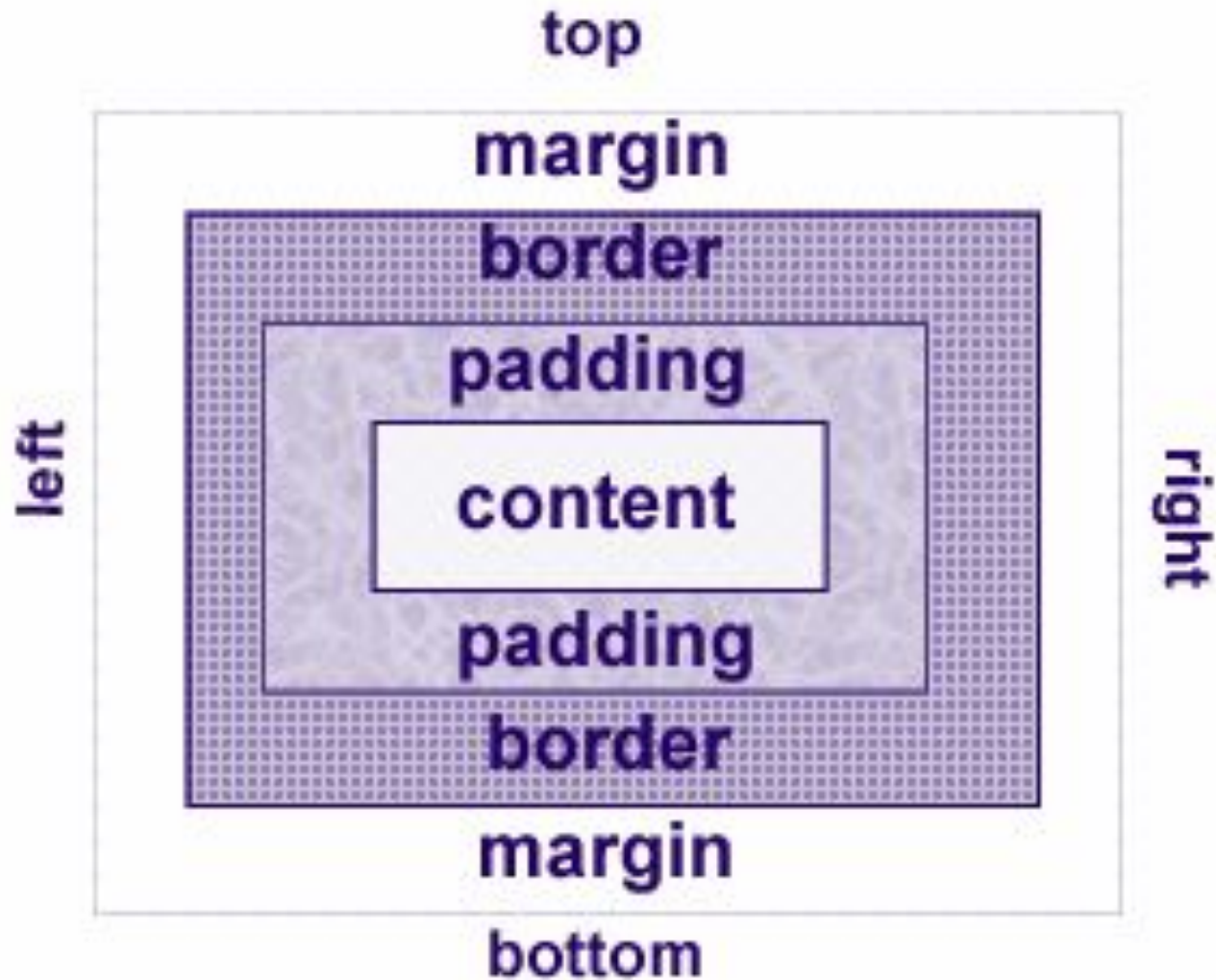
# Learning Outcomes (2 of 2)

- 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

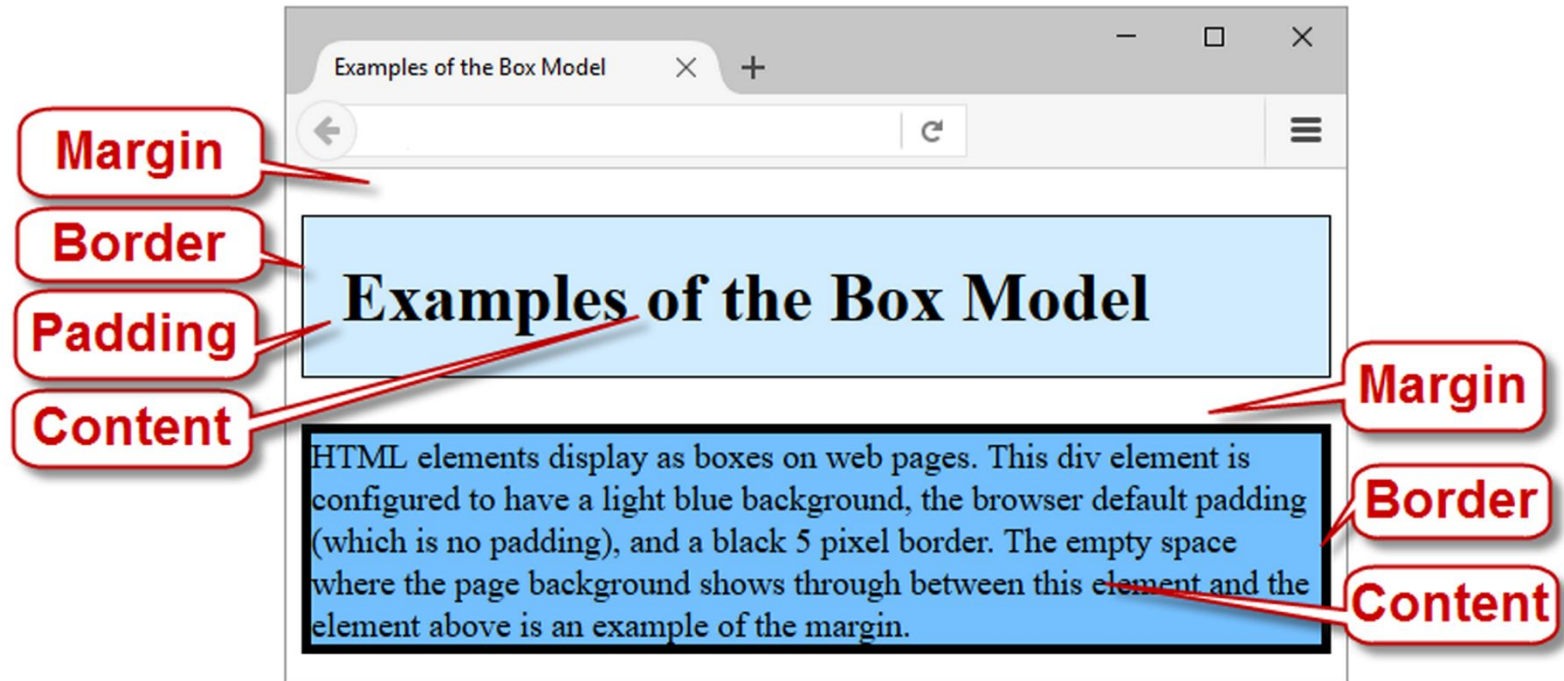
# CSS Width and Height Properties

- width property
  - Configures the width of an element's content
- min-width property
  - Configures minimum width of an element
  - `body{width : 80%; min-width: 750px}`
- max-width property
  - Configures the maximum width of an element
- height property
  - Configures the height of an element
  - `h1{height : 150px; line-height: 150px}`

# The CSS box model



# Example of the Box Model



# Configure Margin with CSS

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

```
h1 { margin: 0; }
```

```
h1 { margin: 50px 10px; }
```

```
h1 { margin: 10px 100px 70px; }
```

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

# Configure Padding with CSS

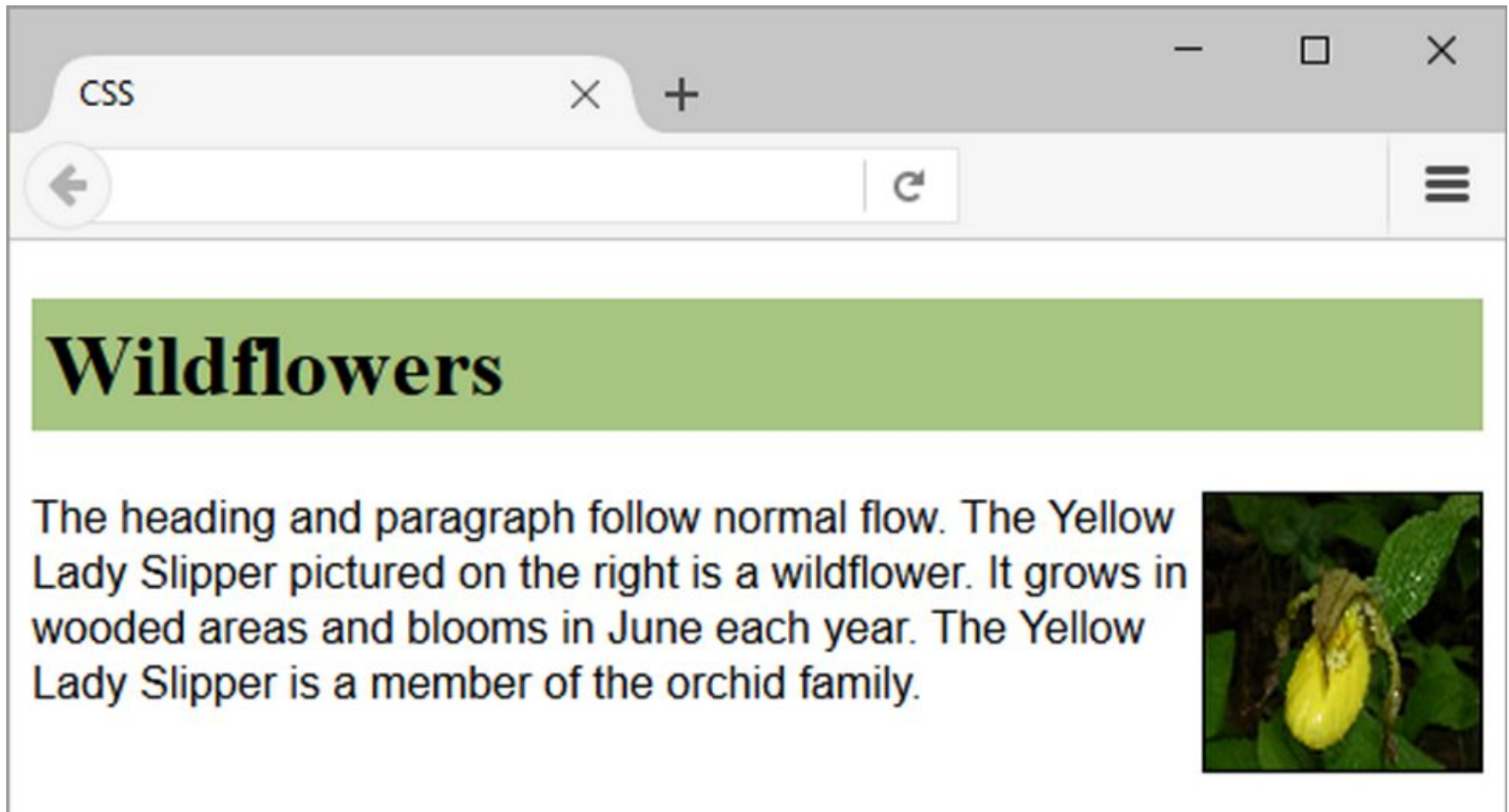
- The padding property
  - padding-top, padding-right, padding-left, padding-bottom
- Configures empty space between the content of the HTML element (such as text) and the border
- Examples

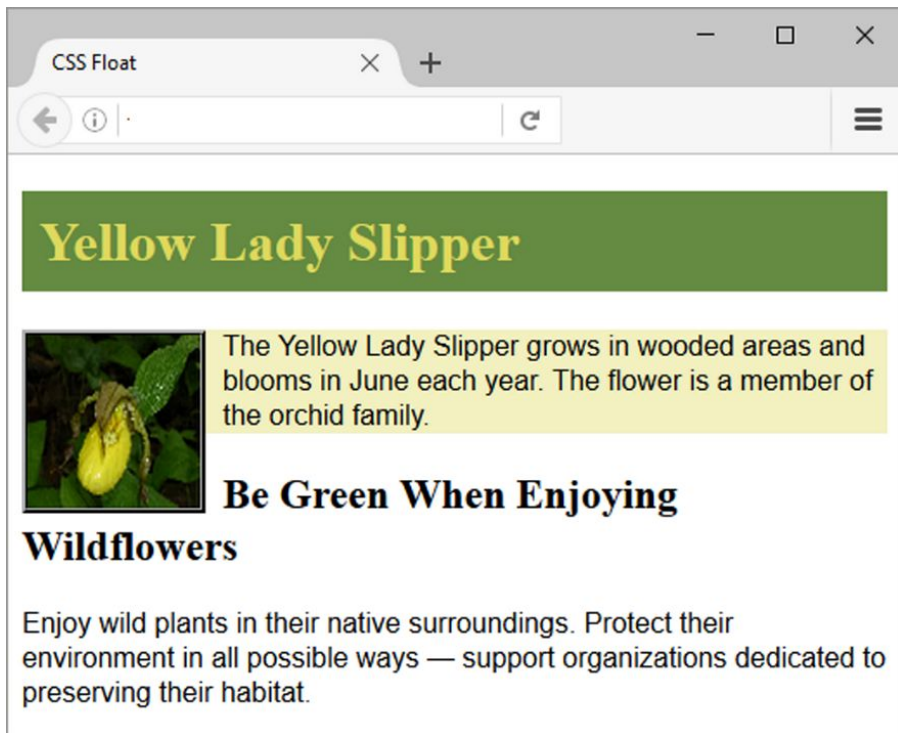
```
h1 { padding: 0; }  
h1 { padding: 20px 10px; }  
h1 { padding: 10px 30px 20px; }  
h1 { padding: 20px 30px 0 30px; }
```





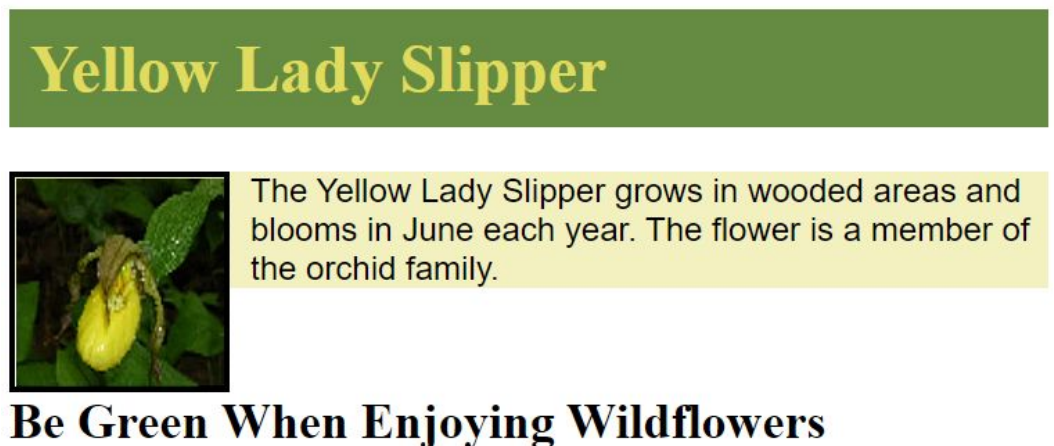
# Example of a float element – float right





H2 is displayed in normal flow.

clear Property



# 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

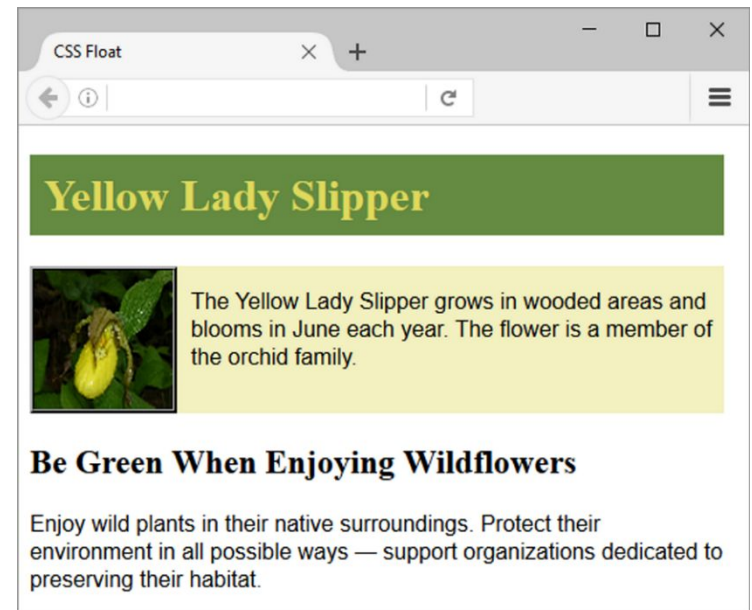
Values are auto, hidden, and scroll

```
div { background-color: #F3F1BF;
```

```
    overflow: auto;
```

```
    width: 100%;
```

```
}
```



## Checkpoint (1 of 2)

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?

# 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.

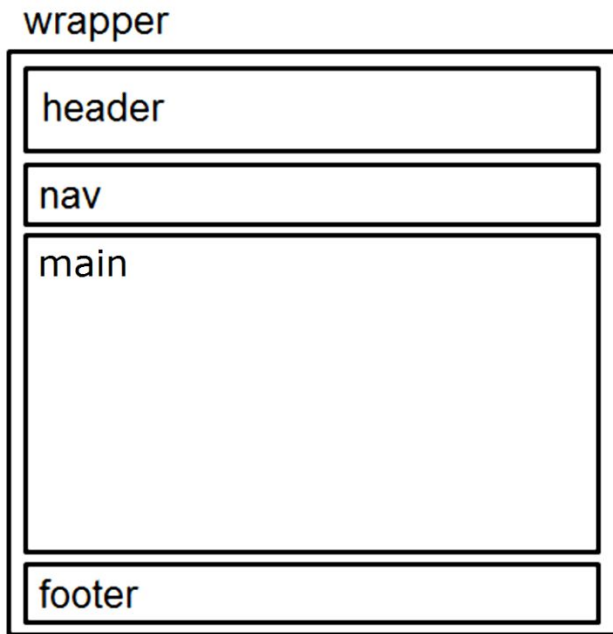
Use the universal selector (\*) to apply this to all the element on the page

Example:

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

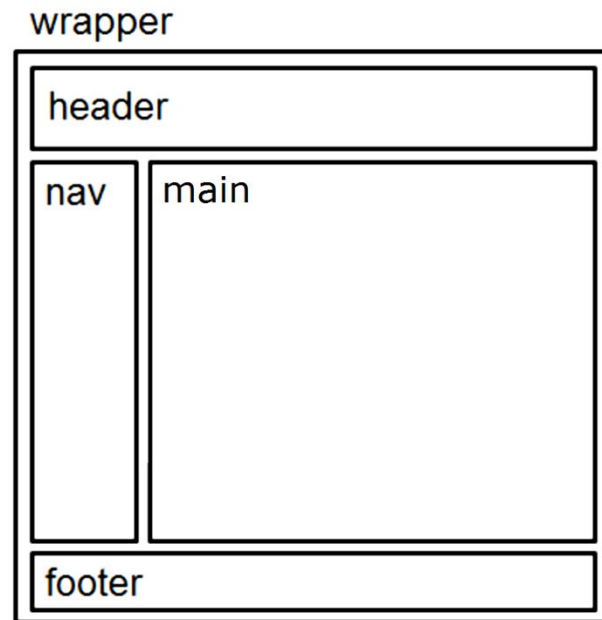
# Page Layout Single Column -> Two Column

## Single Column Wireframe



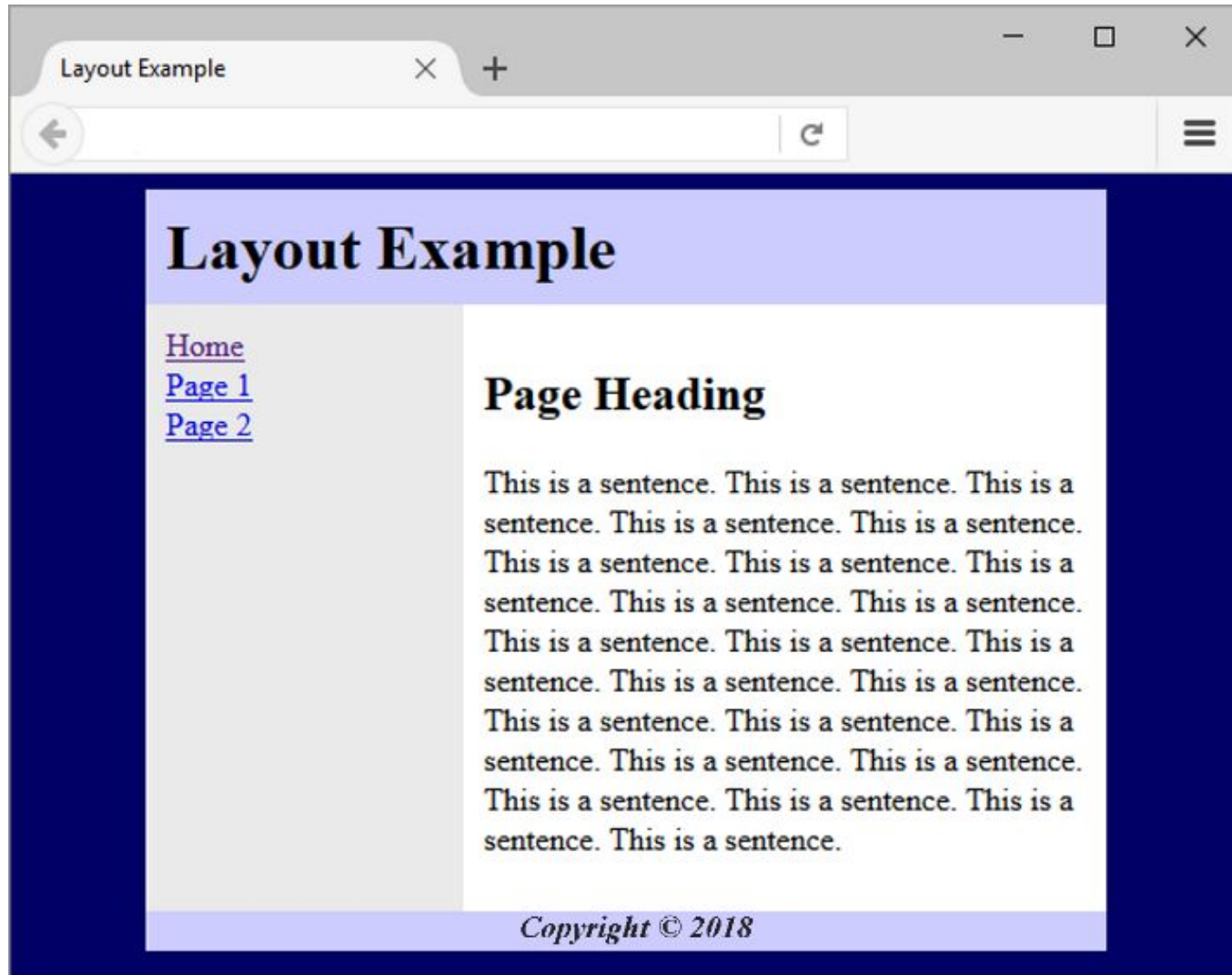
**Figure 6.18** Single-column layout.

## Two Column Wireframe



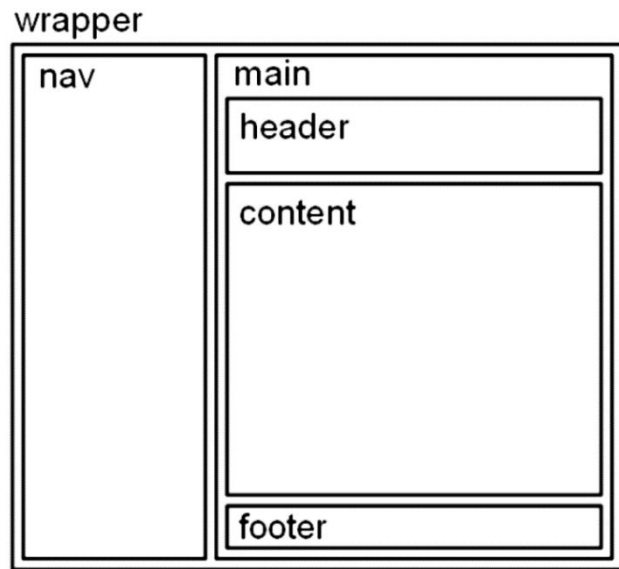
**Figure 6.19** Two-column layout.

# Two-column layout





# CSS Page Layout Two Columns (left nav)



Alternate wireframe

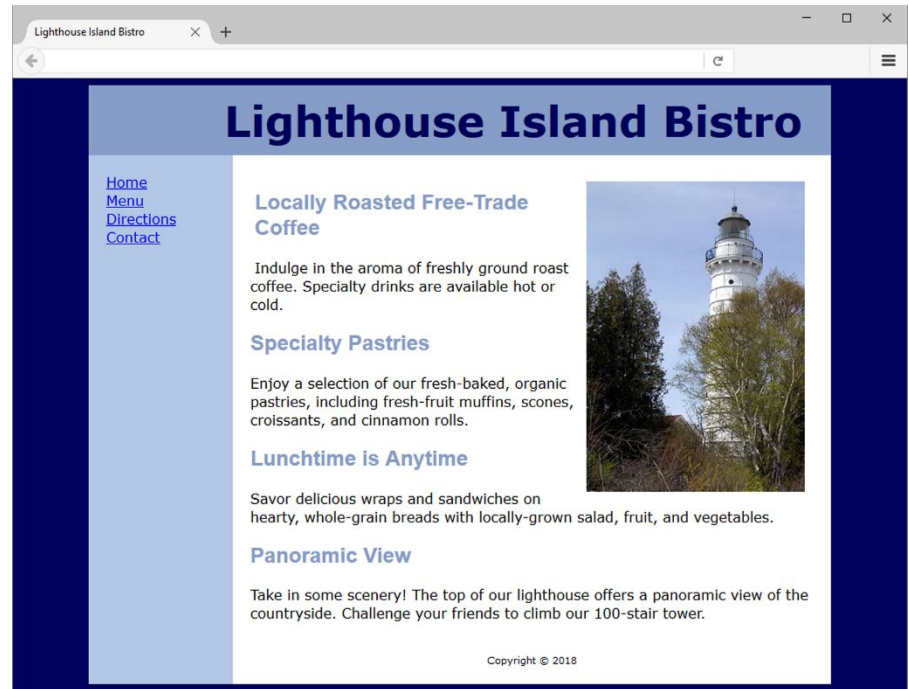


Figure 6.25 Page with alternate layout

# 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>`.

# Vertical navigation

CSS removes the list marker and underline:

```
nav ul { list-style-type: none; }
```

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

- 
- [Home](#)
  - [Menu](#)
  - [Directions](#)
  - [Contact](#)

Navigation in an  
unordered list



[Home](#)  
[Menu](#)  
[Directions](#)  
[Contact](#)

The CSS text-decoration  
property has been applied

# 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>
```

# Horizontal Navigation

[Home](#) [Menu](#) [Directions](#) [Contact](#)

```
<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>
```

```
nav ul { list-style-type: none;}
```

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

```
        padding-right: 10px; }
```

```
nav li { display: inline; }
```

# CSS Pseudo-classes (1 of 2)

- 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;}
```

# CSS Pseudo-classes (2 of 2)

```
a:link { color: #ff0000; }
```

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

```
color: #000066; }
```

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** Using the `hover` pseudo-class

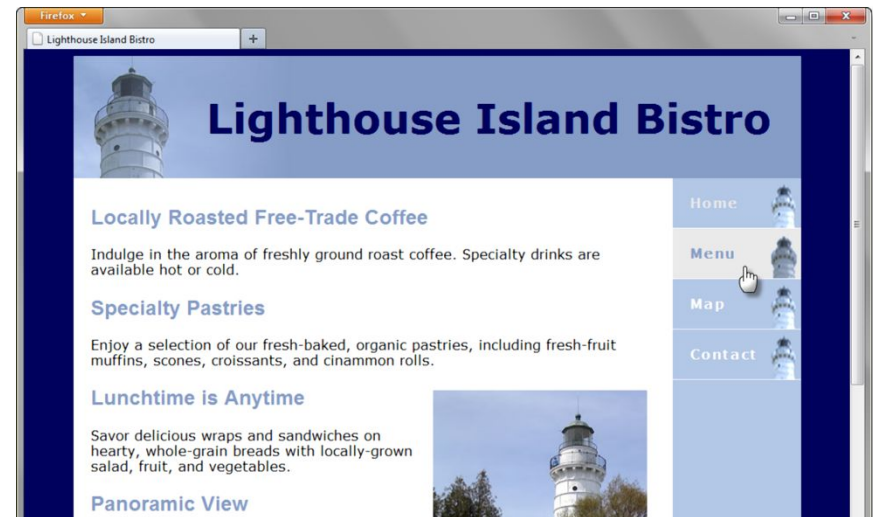
# CSS Sprites

## Sprite

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

```
nav a { background-image: url(sprites.gif);  
        background-repeat: no-repeat;  
        background-position: right 0;  
}
```

```
nav a:hover { color: #869DC7;  
              background-color: #EAEAEA;  
              background-position: right -100px;  
}
```



Sprites in action



# 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">
```

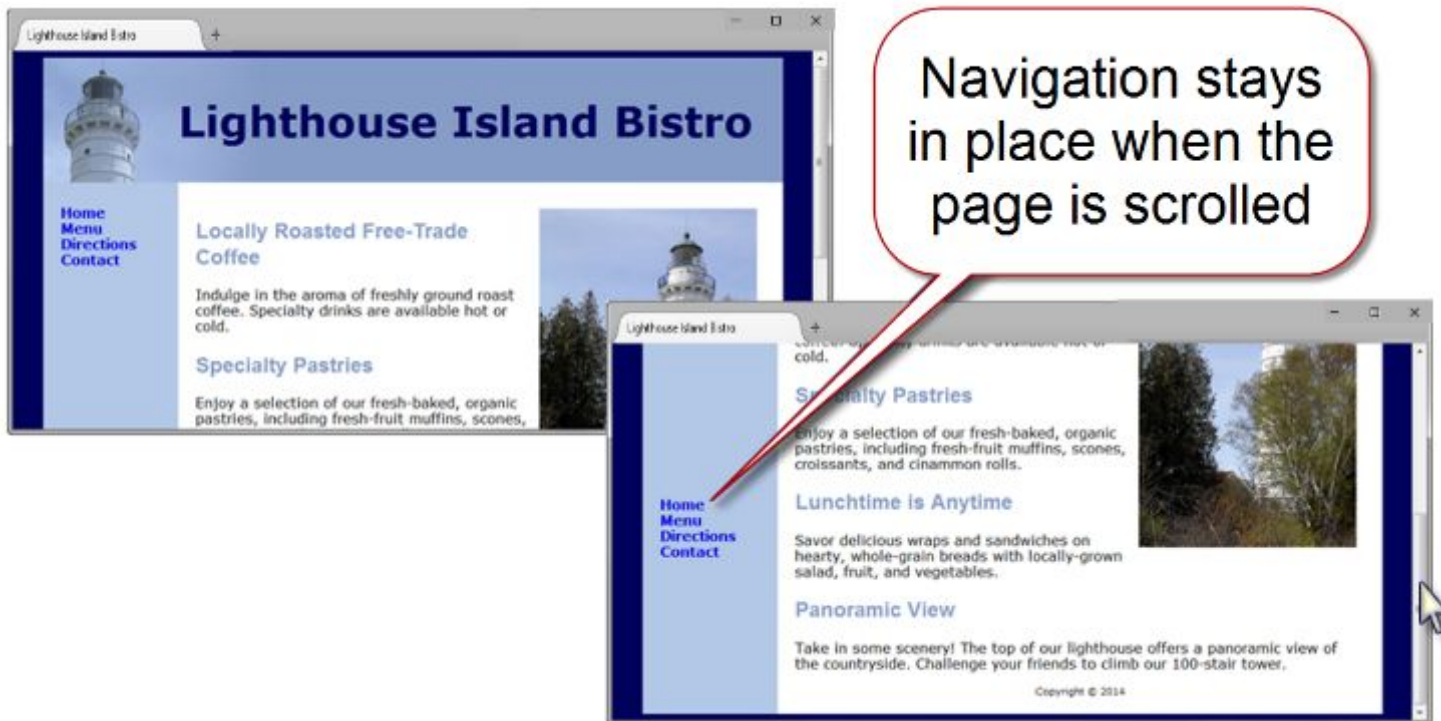
# Position Property

**Table 6.8** The position Property

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

# Fixed Positioning

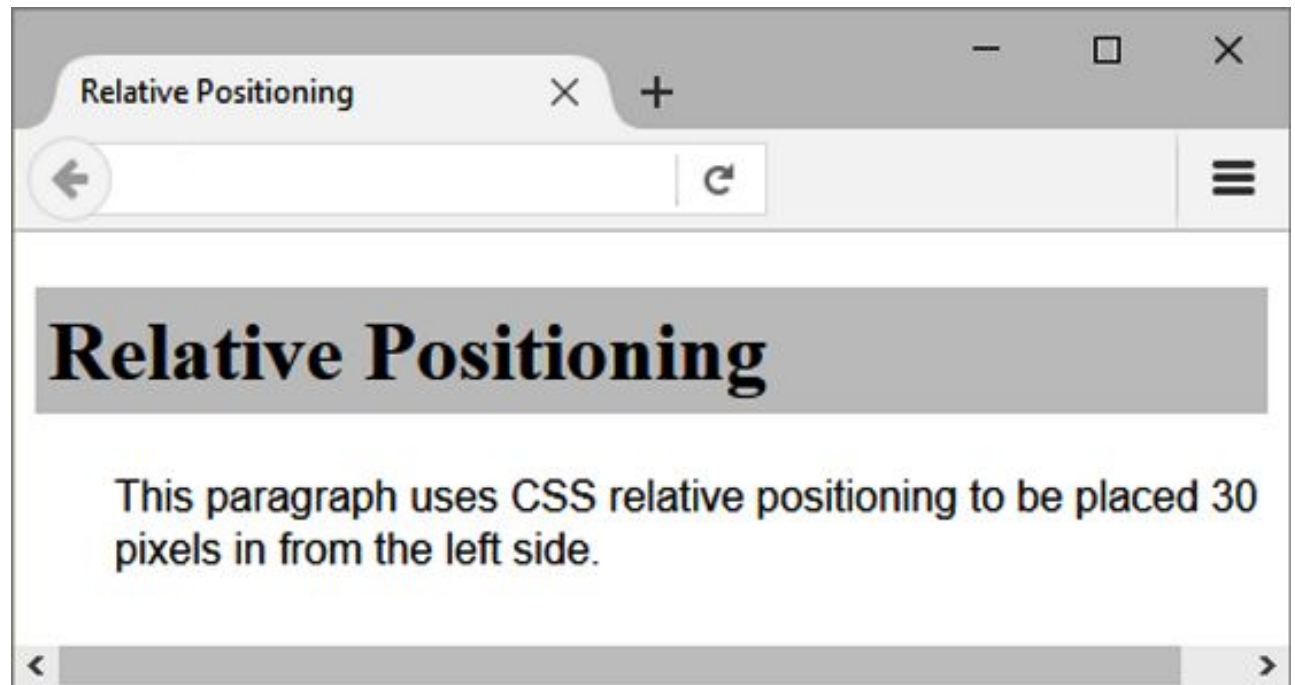
```
nav { position: fixed; }
```



The navigation is configured with fixed positioning

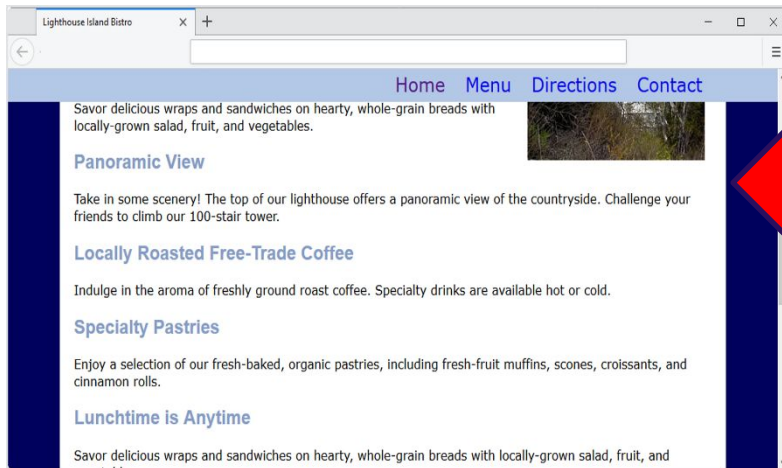
# Relative Positioning

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



The paragraph is configured using relative positioning

# Sticky Positioning



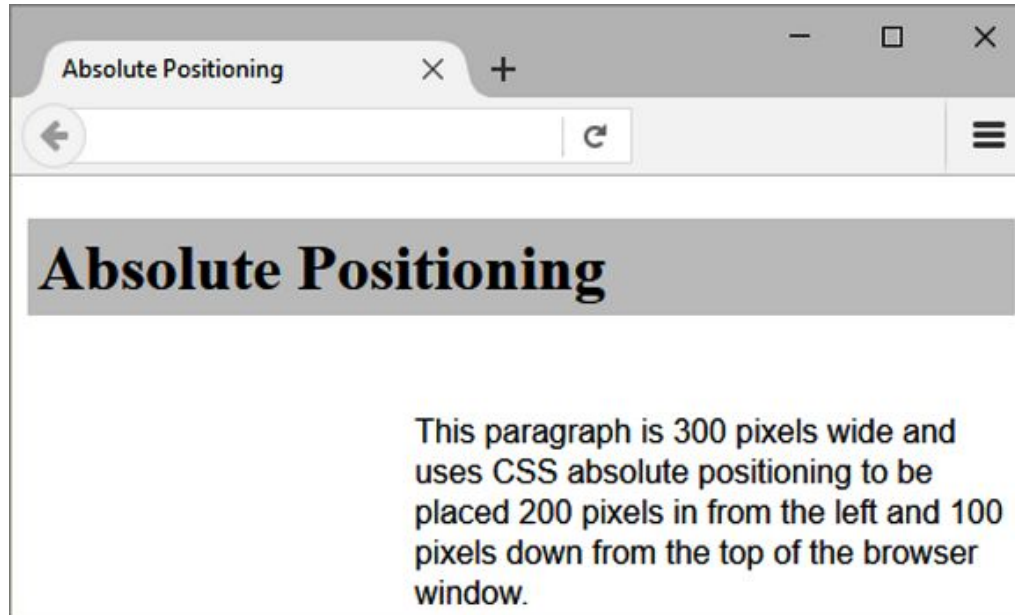
“Sticky” Navigation Bar

The navigation bar stays in place while the content is scrolled.

```
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



- Precisely specifies the location of an element outside of normal flow 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

```
img {  
  position: absolute;  
  left: 0px;  
  top: 0px;  
  z-index: -1;  
}
```

```

```

```
<p>Because the image has a z-index of -1,  
it will be placed behind the heading.</p>
```



## The z-index Property

Because the image has a z-index of -1, it will be placed behind the heading.

With Z index



## index Property

image has a z-index of -1, it will be placed behind the heading.

Without Z index

# 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.

`<a href="#top">Back to Top</a>`

*Note the use of the # in the anchor tag!*



# Single Page Website

- 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

Top Fixed Navigation
Home “page” with hero image
Tours “page”
Tours hero image
Rentals “page”
Rentals hero image
Contact “page”
Contact hero image
Bottom Fixed Footer

# 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;

# 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.