

Introduction to Cascading Style Sheets™ (CSS): Part I

4

Fashions fade, style is eternal.

—Yves Saint Laurent

How liberating to work in the margins, outside a central perception.

—Don DeLillo

Objectives

In this chapter you'll:

- Control a website's appearance with style sheets.
- Use a style sheet to give all the pages of a website the same look and feel.
- Use the `class` attribute to apply styles.
- Specify the precise font, size, color and other properties of displayed text.
- Specify element backgrounds and colors.
- Understand the box model and how to control margins, borders and padding.
- Use style sheets to separate presentation from content.

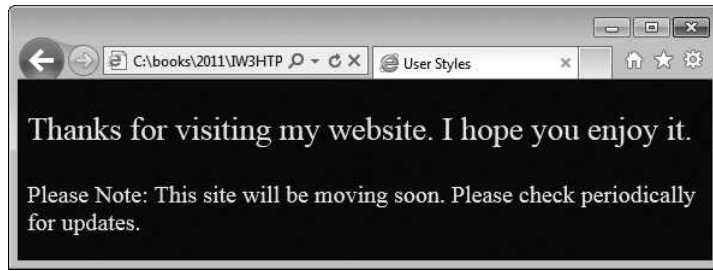


Fig. 4.23 | User style sheet applied with `em` measurement.

4.14 Web Resources

<http://www.deitel.com/css3>

The Deitel CSS3 Resource Center contains links to some of the best CSS3 information on the web. There you'll find categorized links to tutorials, references, code examples, demos, videos, and more. Check out the demos section for more advanced examples of layouts, menus and other web-page components.

Summary

Section 4.1 Introduction

- Cascading Style Sheets™ 3 (CSS3; p. 106) allows you to specify the presentation of elements on a web page (e.g., fonts, spacing, sizes, colors, positioning) separately from the structure and content of the document (section headers, body text, links, etc.).
- This separation of structure from presentation (p. 106) simplifies maintaining and modifying web pages, especially on large-scale websites.

Section 4.2 Inline Styles

- An inline style (p. 106) allows you to declare a style for an individual element by using the `style` attribute (p. 106) in the element's start tag.
- Each CSS property (such as `font-size`, p. 107) is followed by a colon and the value of the attribute. Multiple property declarations are separated by a semicolon.
- The `color` property (p. 107) sets text color. Hexadecimal codes or color names may be used.

Section 4.3 Embedded Style Sheets

- Embedded style sheets (p. 108) enable you to embed an entire CSS3 document in an HTML5 document's head section.
- Styles that are placed in a `style` element use selectors (p. 109) to apply style elements throughout the entire document body.
- An `em` element indicates that its contents should be emphasized. Browsers usually render `em` elements in an italic font.

- `style` element attribute type specifies the MIME type (the specific encoding format, p. 109) of the style sheet. Style sheets use `text/css`.
- Each rule body (p. 109) in a style sheet begins and ends with a curly brace (`{` and `}`).
- The `font-weight` property (p. 110) specifies the “boldness” of text. Possible values are `bold`, `normal` (the default), `bolder` (bolder than bold text) and `lighter` (lighter than normal text).
- Boldness also can be specified with multiples of 100, from 100 to 900. Text specified as `normal` is equivalent to 400, and `bold` text is equivalent to 700.
- Style-class declarations are preceded by a period and are applied to elements of the specific class. The `class` attribute (p. 111) applies a style class to an element.
- The CSS rules in a style sheet use the same format as inline styles.
- The `background-color` attribute specifies the background color of the element.
- The `font-family` property (p. 110) names a specific font that should be displayed. Generic font families allow authors to specify a type of font instead of a specific one, in case a browser does not support a specific font.
- The `font-size` property (p. 110) specifies the size used to render the font.
- You should end a font list with a generic font family (p. 110) name in case the other fonts are not installed on the user’s computer.
- In many cases, the styles applied to an element (the parent or ancestor element, p. 111) also apply to the element’s nested elements (child or descendant elements, p. 111).

Section 4.4 Conflicting Styles

- Styles may be defined by a user, an author or a user agent. A user (p. 111) is a person viewing your web page, you’re the author (p. 111)—the person who writes the document—and the user agent (p. 111) is the program used to render and display the document (e.g., a web browser).
- Styles cascade (hence the term “Cascading Style Sheets,” p. 111), or flow together, such that the ultimate appearance of elements on a page results from combining styles defined in several ways.
- Most styles are inherited from parent elements (p. 111). Styles defined for children (p. 111) have higher specificity (p. 112) and take precedence over the parent’s styles.
- Pseudo-classes (p. 113) give the author access to information that’s not declared in the document, such as whether the mouse is hovering over an element or whether the user has previously clicked (visited) a particular hyperlink. The hover pseudo-class (p. 114) is activated when the user moves the mouse cursor over an element.
- The `text-decoration` property (p. 113) applies decorations to text in an element, such as `underline`, `overline` and `line-through`.
- To apply rules to multiple elements, separate the elements with commas in the style sheet.
- To apply rules only to a certain type of element that’s a child of another type, separate the element names with spaces.
- A pixel is a relative-length measurement (p. 114): It varies in size based on screen resolution. Other relative lengths are `em` (p. 114), `ex` (p. 114) and percentages.
- The other units of measurement available in CSS are absolute-length measurements (p. 114)—that is, units that do not vary in size. These units can be `in` (inches), `cm` (centimeters, p. 114), `mm` (millimeters, p. 114), `pt` (points; 1 pt = 1/72 in, p. 114) or `pc` (picas; 1 pc = 12 pt).

Section 4.5 Linking External Style Sheets

- With external style sheets (i.e., separate documents that contain only CSS rules; p. 114), you can provide a uniform look and feel to an entire website (or to a portion of one).

- When you need to change styles, you need to modify only a single CSS file to make style changes across all the pages that use those styles. This is sometimes known as skinning (p. 114).
- CSS comments (p. 115) may be placed in any type of CSS code (i.e., inline styles, embedded style sheets and external style sheets) and always start with `/*` and end with `*/`.
- Link's `rel` attribute (p. 115) specifies a relationship between two documents (p. 115). For style sheets, the `rel` attribute declares the linked document to be a `stylesheet` (p. 115) for the document. The `type` attribute specifies the MIME type of the related document as `text/css`. The `href` attribute provides the URL for the document containing the style sheet.

Section 4.6 Positioning Elements: Absolute Positioning, z-index

- The CSS `position` property (p. 116) allows absolute positioning (p. 117), which provides greater control over where on a page elements reside. Specifying an element's position as `absolute` removes it from the normal flow of elements on the page and positions it according to distance from the top, left, right or bottom margin of its parent element.
- The `z-index` property (p. 118) allows a developer to layer overlapping elements. Elements that have higher `z-index` values are displayed in front of elements with lower `z-index` values.

Section 4.7 Positioning Elements: Relative Positioning, span

- Unlike absolute positioning, relative positioning keeps elements in the general flow on the page and offsets them by the specified top, left, right or bottom value.
- Element `span` (p. 120) is a grouping element (p. 120)—it does not apply any inherent formatting to its contents. Its primary purpose is to apply CSS rules or `id` attributes to a section of text.
- `span` is an inline-level element (p. 120)—it applies formatting to text without changing the flow of the document. Examples of inline elements include `span`, `img`, `a`, `em` and `strong`.
- The `div` element is also a grouping element, but it's a block-level element. This means it's displayed on its own line and has a virtual box around it. Examples of block-level elements (p. 120) include `div` (p. 120), `p` and heading elements (`h1` through `h6`).

Section 4.8 Backgrounds

- Property `background-image` specifies the URL of the image, in the format `url(fileLocation)`. The property `background-position` (p. 121) places the image on the page using the values top, bottom, center, left and right individually or in combination for vertical and horizontal positioning. You can also position by using lengths.
- The `background-repeat` property (p. 121) controls the tiling of the background image (p. 121). Setting the tiling to `no-repeat` displays one copy of the background image on screen. The `background-repeat` property can be set to `repeat` (the default) to tile the image vertically and horizontally, to `repeat-x` to tile the image only horizontally or to `repeat-y` to tile the image only vertically.
- The `background-attachment` (p. 121) setting `fixed` fixes the image in the position specified by `background-position`. Scrolling the browser window will not move the image from its set position. The default value, `scroll`, moves the image as the user scrolls the window.
- The `text-indent` property (p. 122) indents the first line of text in the element by the specified amount.
- The `font-style` property (p. 122) allows you to set text to `none`, `italic` or `oblique`.

Section 4.9 Element Dimensions

- An element's dimensions can be set with CSS by using properties `height` and `width` (p. 123).
- Text in an element can be centered using `text-align` (p. 123); other values for the `text-align` property are `left` and `right`.

- A problem with setting both vertical and horizontal dimensions of an element is that the content inside the element might sometimes exceed the set boundaries, in which case the element grows to fit the content. You can set the `overflow` property (p. 123) to `scroll`; this setting adds scroll bars if the text overflows the boundaries set for it.

Section 4.10 Box Model and Text Flow

- All block-level HTML5 elements have a virtual box drawn around them, based on what is known as the box model (p. 123).
- When the browser renders elements using the box model, the content of each element is surrounded by padding (p. 123), a border (p. 123) and a margin (p. 123).
- The `border-width` property (p. 124) may be set to any of the CSS lengths or to the predefined value of `thin`, `medium` or `thick`.
- The `border-styles` (p. 124) available are `none`, `hidden`, `dotted`, `dashed`, `solid`, `double`, `groove`, `ridge`, `inset` and `outset`.
- The `border-color` property (p. 124) sets the color used for the border.
- The `class` attribute allows more than one class to be assigned to an element by separating each class name from the next with a space.
- Browsers normally place text and elements on screen in the order in which they appear in the document. Elements can be removed from the normal flow of text. Floating allows you to move an element to one side of the screen; other content in the document will then flow around the floated element.
- CSS uses a box model to render elements on screen. The content of each element is surrounded by padding, a border and margins. The properties of this box are easily adjusted.
- The `margin` property (p. 127) determines the distance between the outside edge of the element's border and any adjacent element.
- Margins for individual sides of an element can be specified by using `margin-top`, `margin-right`, `margin-left` and `margin-bottom`.
- The `padding` property (p. 127) determines the distance between the content inside an element and the inside edge of the border. Padding also can be set for each side of the box by using `padding-top`, `padding-right`, `padding-left` and `padding-bottom`.

Section 4.11 Media Types and Media Queries

- CSS media types (p. 127) allow you to decide what a page should look like depending on the kind of media being used to display the page. The most commonly used for a web page is the `screen` media type (p. 127), which is a standard computer screen.
- A block of styles that applies to all media types is declared by `@media all` and enclosed in curly braces. To create a block of styles that apply to a single media type such as `print`, use `@media print` and enclose the style rules in curly braces.
- Other media types in CSS 2 include `handheld`, `braille`, `aural` and `print`. The `handheld` medium (p. 127) is designed for mobile Internet devices, while `braille` (p. 127) is for machines that can read or print web pages in braille. `aural` styles (p. 127) allow the programmer to give a speech-synthesizing web browser more information about the content of the web page. The `print` media type (p. 127) affects a web page's appearance when it's printed.
- Media queries (p. 130) allow you to format your content to specific output devices. Media queries include a media type and expressions that check the devices' media features (p. 130).

Section 4.12 Drop-Down Menus

- The `:hover` pseudo-class is used to apply styles to an element when the mouse cursor is over it.

- The `display` property (p. 130) allows you to decide whether an element is displayed as a block element or inline element or not rendered at all (none).

Section 4.13 (Optional) User Style Sheets

- Users can define their own user style sheets (p. 132) to format pages based on their preferences.
- Absolute font-size measurements override user style sheets, while relative font sizes will yield to a user-defined style.
- If the user defines font size in a user style sheet, the author style (p. 133) has a higher precedence and overrides the user style.

Self-Review Exercises

- 4.1** Assume that the size of the base font on a system is 12 points.
- How big is a 36-point font in ems?
 - How big is a 9-point font in ems?
 - How big is a 24-point font in picas?
 - How big is a 12-point font in inches?
 - How big is a 1-inch font in picas?
- 4.2** Fill in the blanks in the following statements:
- Using the _____ element allows you to use external style sheets in your pages.
 - To apply a CSS rule to more than one element at a time, separate the element names with a(n) _____.
 - Pixels are a(n) _____-length measurement unit.
 - The _____ pseudo-class is activated when the user moves the mouse cursor over the specified element.
 - Setting the _____ property to `scroll` provides a mechanism for creating scrollable content without compromising an element's dimensions.
 - _____ is a generic inline element that applies no inherent formatting, and _____ is a generic block-level element that applies no inherent formatting.
 - Setting property `background-repeat` to _____ tiles the specified background-image vertically.
 - To begin a block of styles that applies only to the print media type, you use the declaration _____ `print`, followed by an opening curly brace (`{`).
 - The _____ property allows you to indent the first line of text in an element.
 - The three components of the box model are the _____, _____ and _____.

Answers to Self-Review Exercises

- 4.1** a) 3 ems. b) 0.75 ems. c) 2 picas. d) 1/6 inch. e) 6 picas.
- 4.2** a) `link`. b) comma. c) relative. d) `:hover`. e) `overflow`. f) `span`, `div`. g) `repeat-y`. h) `@media`. i) `text-indent`. j) padding, border, margin.

Exercises

- 4.3** Write a CSS rule that makes all text 1.5 times larger than the base font of the system and colors the text red.
- 4.4** Write a CSS rule that places a background image halfway down the page, tiling it horizontally. The image should remain in place when the user scrolls up or down.

- 4.5** Write a CSS rule that gives all h1 and h2 elements a padding of 0.5 ems, a dashed border style and a margin of 0.5 ems.
- 4.6** Write a CSS rule that changes the color of all elements containing attribute `class = "green-Move"` to green and shifts them down 25 pixels and right 15 pixels.
- 4.7** Make a layout template that contains a header and two paragraphs. Use `float` to line up the two paragraphs as columns side by side. Give the header and two paragraphs a border and/or a background color so you can see where they are.
- 4.8** Add an *embedded style sheet* to the HTML5 document in Fig. 2.3. The style sheet should contain a rule that displays h1 elements in blue. In addition, create a rule that displays all links in blue without underlining them. When the mouse hovers over a link, change the link's background color to yellow.
- 4.9** Make a navigation button using a `div` with a link inside it. Give it a border, background, and text color, and make them change when the user hovers the mouse over the button. Use an external style sheet. Make sure your style sheet validates at <http://jigsaw.w3.org/css-validator/>. Note that some warnings may be unavoidable, but your CSS should have no errors.

5

Introduction to Cascading Style Sheets™ (CSS): Part 2

Art is when things appear rounded.

—Maurice Denis

In matters of style, swim with the current; in matters of principle, stand like a rock.

—Thomas Jefferson

Everything that we see is a shadow cast by that which we do not see.

—Martin Luther King, Jr.

Objectives

In this chapter you'll:

- Add text shadows and text-stroke effects.
- Create rounded corners.
- Add shadows to elements.
- Create linear and radial gradients, and reflections.
- Create animations, transitions and transformations.
- Use multiple background images and image borders.
- Create a multicolumn layout.
- Use flexible box model layout and `:nth-child` selectors.
- Use the `@font-face` rule to specify fonts for a web page.
- Use RGBA and HSLA colors.
- Use vendor prefixes.
- Use media queries to customize content to fit various screen sizes.



Summary

Section 5.2 Text Shadows

- The CSS3 `text-shadow` property (p. 143) makes it easy to add a text-shadow effect to any text. The shadow's horizontal offset is the number of pixels that the text-shadow will appear to the left or the right of the text. A negative value moves the text-shadow to the left; a positive value moves it to the right. The vertical offset is the number of pixels that the text-shadow will be shifted up or down from the text. A negative value moves the shadow up, whereas a positive value moves it down.
- The blur radius (p. 143) has a value of 0 (no shadow) or greater.

Section 5.3 Rounded Corners

- The `border-radius` property (p. 144) adds rounded corners (p. 144) to any element.

Section 5.4 Color

- RGBA (Red, Green, Blue, Alpha, p. 145) gives you greater control over the exact colors in your web pages. The value for each color—red, green and blue—can range from 0 to 255. The alpha value—which represents opacity—can be any value in the range 0.0 (fully transparent) through 1.0 (fully opaque).
- CSS3 also allows you to express color using HSLA (hue, saturation, lightness, alpha) values (p. 146).
- The hue is a color or shade expressed as a value from 0 to 359 representing the degrees on a color wheel (a wheel is 360 degrees). The colors on the wheel progress in the order of the colors of the rainbow—red, orange, yellow, green, blue, indigo and violet.
- The saturation (p. 146)—the intensity of the hue—is expressed as a percentage, where 100% is fully saturated (the full color) and 0% is gray.
- Lightness (p. 146)—the intensity of light or luminance of the hue—is also expressed as a percentage. A lightness of 50% is the actual hue. If you decrease the amount of light to 0%, the color appears completely dark (black). If you increase the amount of light to 100%, the color appears completely light (white).

Section 5.5 Box Shadows

- The `box-shadow` property (p. 146) adds a shadow to an element.
- The horizontal offset of the shadow defines the number of pixels that the box-shadow will appear to the left or the right of the box. The vertical offset of the shadow defines the number of pixels the box-shadow will be shifted up or down from the box.
- The blur radius of the shadow can have a value of 0 (no shadow) or greater.

Section 5.6 Linear Gradients; Introducing Vendor Prefixes

- Linear gradients (p. 148) are a type of image that gradually transitions from one color to the next horizontally, vertically or diagonally.
- You can transition between as many colors as you like and specify the points at which to change colors, called `color-stops` (p. 148), represented in pixels or percentages along the so-called gradient line.
- You can use gradients in any property that accepts an image.
- Browsers currently implement gradients differently, so you'll need vendor prefixes and different syntax for each browser.

- Vendor prefixes (e.g., `-webkit-` and `-moz-`, p. 151) are used for properties that are still being finalized in the CSS specification but have already been implemented in various browsers.
- Prefixes are not available for every browser or for every property.
- It's good practice to include the multiple prefixes when they're available so that your pages render properly in the various browsers.
- Always place vendor-prefixed styles before the nonprefixed version. The last version of the style that a given browser supports takes precedence and will be used by the browser.

Section 5.7 Radial Gradients

- Radial gradients (p. 151) are similar to linear gradients, but the color changes gradually from an inner circle (the start) to an outer circle (the end).
- The `radial-gradient` property (p. 151) has three values. The first is the position of the start of the radial gradient (center). Other possible values for the position include `top`, `bottom`, `left` and `right`. The second value is the start color, and the third is the end color.
- Other than the vendor prefixes, the syntax of the gradient is identical for WebKit browsers, Mozilla Firefox and the standard CSS3 `radial-gradient`.

Section 5.8 (Optional: WebKit Only) Text Stroke

- The `-webkit-text-stroke` property (p. 153) is a nonstandard property for WebKit-based browsers that allows you to add an outline (text stroke) around text. The `-webkit-text-stroke` property has two values—the thickness of the outline and the color of the text stroke.

Section 5.9 Multiple Background Images

- CSS3 allows you to add multiple background images (p. 153) to an element.
- We specify each image's placement using property `background-position`. The comma-separated list of values matches the order of the comma-separated list of images in the `background-image` property.
- The `background-origin` (p. 154) determines where each image is placed using the box model.

Section 5.10 (Optional: WebKit Only) Reflections

- The `-webkit-box-reflect` property (p. 155) allows you to add a simple reflection (p. 155) of an image. Like `-webkit-text-stroke`, this is a nonstandard property that's available only in WebKit-based browsers for now.
- The property's first value is the *direction* of the reflection. The direction value may be `above`, `below`, `left`, or `right`.
- The second value is the *offset*, which determines the space between the image and its reflection.
- Optionally, you can specify a gradient to apply to the reflection.

Section 5.11 Image Borders

- The CSS3 `border-image` property (p. 156) uses images to place a border around any element.
- The `border-width` is the thickness of the border being placed around the element. The width is the width of the entire rectangular border.
- The `border-image-source` (p. 157) is the URL of the image to use in the border.
- The `border-image-slice` (p. 158) specifies the inward offsets from the top, right, bottom and left sides of the image.
- The `border-image-slice` divides the image into nine regions: four corners, four sides and a middle, which is transparent unless otherwise specified. You may not use negative values.

- We can express the `border-image-slice` in just two values, in which case the first value represents the top and bottom, and the second value the left and right.
- The `border-image-slice` may be expressed in pixels or percentages.
- `border-image-repeat` (p. 158) specifies how the regions of the border image are scaled and tiled (repeated). By indicating `stretch` just once, we create a border that will stretch the top, right, bottom and left regions to fit the area.
- You may specify two values for the `border-image-repeat` property. For example, if we specified `stretch repeat`, the top and bottom regions of the image border would be stretched, and the right and left regions of the border would be repeated (i.e., tiled) to fit the space, using partial tiles to fill the excess space.
- Other possible values for the `border-image-repeat` property include `round` and `space`. If you specify `round`, the regions are repeated using only whole tiles, and the border image is scaled to fit the area. If you specify `space`, the regions are repeated to fill the area using only whole tiles, and any excess space is distributed evenly around the tiles.

Section 5.12 Animation; Selectors

- The `animation` property (p. 161) allows you to represent several animation properties in a shorthand notation, rather than specifying each animation property separately.
- The `animation-name` (p. 161) represents the name of the animation. This name associates the animation with the keyframes that define various properties of the element being animated at different stages of the animation.
- The `animation-timing-function` (p. 161) determines how the animation progresses in one cycle of its duration. Possible values include `linear`, `ease`, `ease-in`, `ease-out`, `ease-in-out`, `cubic-bezier`. The value `linear` specifies that the animation will move at the same speed from start to finish. The default value, `ease`, starts slowly, increases speed, then ends slowly. The `ease-in` value starts slowly, then speeds up, whereas the `ease-out` value starts faster, then slows down. The `ease-in-out` starts and ends slowly. Finally, the `cubic-bezier` value allows you to customize the timing function with four values between 0 and 1, such as `cubic-bezier(1,0,0,1)`.
- The `animation-duration` (p. 161) specifies the time in seconds (s) or milliseconds (ms) that the animation takes to complete one iteration. The default duration is 0.
- The `animation-delay` (p. 161) specifies the number of seconds or milliseconds after the page loads before the animation begins. The default value is 0. If the `animation-delay` is negative, such as `-3s`, the animation begins three seconds into its cycle.
- The `animation-iteration-count` (p. 161) specifies the number of times the animation will run. The default is 1. You may use the value `infinite` to repeat the animation continuously.
- The `animation-direction` (p. 161) specifies the direction in which the animation will run. The value `alternate` used here specifies that the animation will run in alternating directions. The default value, `normal`, would run the animation in the same direction for each cycle.
- The shorthand `animation` property cannot be used with the `animation-play-state` property (p. 161)—it must be specified separately. If you do not include the `animation-play-state`, which specifies whether the animation is paused or running, it defaults to running.
- For the element being animated, the `@keyframes` rule (p. 161) defines the element's properties that will change during the animation, the values to which those properties will change, and when they'll change.
- The `@keyframes` rule is followed by the name of the animation to which the keyframes are applied. Rules (p. 162) consist of one or more selectors (p. 162) followed by a declaration block (p. 162) in curly braces (`{}`).

- Selectors enable you to apply styles to elements of a particular type or attribute.
- A declaration block consists of one or more declarations, each of which includes the property name followed by a colon (:), a value and a semicolon (;). You may include multiple declarations in a declaration block.

Section 5.13 Transitions and Transformations

- With CSS3 transitions (p. 162), you can change an element's style over a specified duration.
- CSS3 transformations (p. 162) allow you to move, rotate, scale and skew elements.
- Transitions are similar in concept to animations, but transitions allow you to specify only the starting and ending values of the CSS properties being changed. An animation's keyframes enable you to control intermediate states throughout the animation's duration.
- For each property that will change, the transition property (p. 162) specifies the duration of that change.
- As of CSS3, the :hover pseudo-class now works with any element.
- The transform property (p. 162) uses transformation functions (p. 163), such as rotate (p. 163) and scale (p. 164), to perform the transformations.
- The rotate transformation function receives number of degrees. Negative values cause the element to rotate left. A value of 720deg would cause the element to rotate clockwise twice.
- The scale transformation function specifies how to scale the width and height. The value 1 represents the original width or original height, so values greater than 1 increase the size and values less than 1 decrease the size.
- CSS3 transformations also allow you to skew (p. 164) block-level elements, slanting them at an angle either horizontally (skewX) or vertically (skewY).
- The transition-duration is the amount of time it takes to complete the transition.
- The transition-timing-function determines how the transition progresses in one cycle of its duration.

Section 5.14 Downloading Web Fonts and the @font-face Rule

- Using the @font-face rule (p. 166), you can specify fonts for a web page, even if they're not installed on the user's system. Downloadable fonts help ensure a uniform look across client sites.
- You can find numerous free, open-source web fonts at <http://www.google.com/webfonts>. Make sure the fonts you get from other sources have no legal encumbrances.
- The @media screen rule specifies that the font will be used when the document is rendered on a computer screen.
- The @font-face rule includes the font-family, font-style and font-weight. Multiple fonts can be specified with varying styles and weights. The @font-face rule also includes the font's location.

Section 5.15 Flexible Box Layout Module and :nth-child Selectors

- Flexible Box Layout Module (FBLM, p. 168) makes it easy to align the contents of boxes, change their size, change their order dynamically, and lay out the contents in any direction.
- The box-orient property (p. 168) specifies the orientation of the box layout. The default value is horizontal. You can also use vertical.
- In CSS3, you can use selectors to easily style attributes. For example, you can select every other row in a table and change the background color to blue, making the table easier to read. You can also use selectors to enable or disable input elements.

- We use `:nth-child` selectors (p. 171) to select each of the for the four `div` elements in the `flex-box` `div` to style.
- `div:nth-child(1)` selects the `div` element that's the first child of its parent and applies the specified style. Similarly, `div:nth-child(2)` selects the `div` element that's the second child of its parent, `div:nth-child(3)` selects the third child of its parent, and `div:nth-child(4)` selects the fourth child of its parent.
- Setting the `overflow` to `hidden` hides any text that does not fit in the specified paragraph height.

Section 5.16 Multicolumn Layout

- CSS3 allows you to easily create multicolumn layouts (p. 171) using the `column-count` property (p. 171).
- The `column-gap` property (p. 171) specifies the spacing between columns.
- Add lines between columns using the `column-rule` property (p. 171).
- Resizing your browser window changes the width of the columns to fit the three-column layout in the browser.

Section 5.17 Media Queries

- With CSS3 media queries you can determine the finer attributes of the media on which the user is viewing the page, such as the length and width of the viewing area on the screen, to customize your presentation.
- The `@media` rule (p. 174) is used to determine the type and size of device on which the page is rendered. When the browser looks at the rule, the result is either true or false. The rule's styles are applied only if the result is true.

Self-Review Exercises

- 5.1** Fill in the blanks in the following statements:
- The _____ property makes it easy to add a text shadow effect to any text.
 - The _____ property allows you to add rounded corners to any element.
 - CSS3 includes two new ways to express color—_____ and _____.
 - The _____ defines the number of pixels that the box-shadow will appear to the left or the right of the box.
 - _____ are similar to linear gradients, but the color changes gradually from an inner circle (the start) to an outer circle (the end).
 - The _____ divides the image into nine regions: four corners, four sides and a middle, which is transparent unless otherwise specified.
 - The `animation-timing-function` determines how the animation progresses in one cycle of its duration. Possible values include _____, _____, _____, _____, _____ and _____.
 - For the element being animated, the _____ defines the element's properties that will change during the animation, the values to which those properties will change, and when they'll change.
 - _____ are similar in concept to animations, but they allow you to specify only the starting and ending values of the CSS properties being changed. An animation's keyframes enable you to control intermediate states throughout the animation's duration.
 - CSS3 _____ allow you to move, rotate, scale and skew elements.
 - _____ consist of one or more selectors followed by a declaration block in curly braces `{}`.
 - In CSS3, you can use _____ to easily style attributes.

- 5.2** State whether each of the following is *true* or *false*. If *false*, explain why.
- The `@font-face` rule specifies that an embedded font will be used when the document is rendered on a computer screen.
 - You can use gradients in any property that accepts an image.
 - A horizontal gradient gradually changes from top to bottom.
 - You can add lines between columns using the `column-gap` property.
 - The `@media` rule determines the type and size of device on which the page is rendered. When the browser looks at the rule, the result is either true or false. The rule's styles are applied only if the result is false.
 - To add multiple background images to an element, use the `background-position` to specify where each image is placed using the box model.

Answers to Self-Review Exercises

5.1 a) `text-shadow`. b) `border-radius`. c) RGBA and HSLA. d) `horizontalOffset`. e) Radial gradients. f) `border-image-slice`. g) `linear`, `ease`, `ease-in`, `ease-out`, `ease-in-out`, `cubic-bezier`. h) `@keyframes` rule. i) Transitions. j) transformations. k) Rules. l) selectors.

5.2 a) False. The `@media screen` rule specifies that an embedded font will be used when the document is rendered on a computer screen. b) True. c) False. A horizontal gradient gradually changes from left to right. d) False. You can add lines between columns using the `column-rule` property. e) The `@media` rule's styles are applied only if the result is true. f) False. The `background-origin` specifies where each image is placed using the box model.

Exercises

For each of the following, build and render a web page that makes the indicated effect(s) appear. Validate your page with the following validators:

- For CSS3: <http://jigsaw.w3.org/css-validator/> (under **More Options > Profile**, select CSS level 3) [Note: Many CSS3 properties will not validate because they're not yet standardized.]
- For HTML5: http://validator.w3.org/#validate_by_upload

Also, test your page with as many as possible of the seven browsers we're using in this book.

5.3 (*Text Shadow*) Create a text shadow on the phrase "New features in CSS3" with a horizontal offset of 2px, a vertical offset of 5px, a blur radius of 6px and a `text-shadow` color `deepskyblue`.

5.4 (*Text Stroke*) Create a text stroke on the phrase "New WebKit features". Make the color of the text `LightBlue`. Use a 3px `Navy` `text-stroke` and set the `font-size` to 700%.

5.5 (*Rounded Corners*) Create three `div` elements, each with a width and height of 100px. On the first element, create slightly rounded corners using a border of 3px `black` and `border-radius` of 10px. On the second element, use a border of 3px `black` and increase the `border-radius` to 50px. On the third, use a border of 3px `black` and increase the `border-radius` to 100px. Make the `background-color` of each element a different color of your choosing. Inside each element, display the value of the `border-radius` in bold text.

5.6 (*Box Shadow*) Create three `div` elements of varying colors, each with a width and height of 200px. On the first box, add a `dimgrey` `box-shadow` with a horizontal offset of 15px, a vertical offset of 15px and a blur radius of 20px. On the second box, add a `dimgrey` `box-shadow` with a horizontal offset of -15px, a vertical offset of -15px and a blur radius of 30px. On the third box, add a `dimgrey` `box-shadow` with a horizontal offset of 15px, a vertical offset of 15px and a blur radius of 10px.

- 5.7** (*Linear Gradient*) Create a div element with a width and height of 500px. Create a diagonal linear gradient using the colors of the rainbow—Red, Orange, Yellow, Green, Blue, Indigo, Violet.
- 5.8** (*Radial Gradient*) Create a div element with a width and height of 500px. Create a radial gradient with three colors. Start the gradient in the bottom-left corner with the colors changing as they move along the gradient line to the right.
- 5.9** (*Animation*) Create an infinite animation of an element moving in a square pattern.
- 5.10** (*Skew*) Modify the skew example in Fig. 5.13 to skew the element top to bottom 30deg, then left to right 30deg, alternating infinitely.
- 5.11** (*Melting Images*) Modify the example in Fig. 5.14 using five pictures. It might be interesting to try pictures of you or a family member at different ages or a landscape at various times. Set the transition-duration to 3s and a transition-timing-function to linear.
- 5.12** (*Multicolumn Text*) Change the format of the example in Fig. 5.17 to two columns, add a sub title and an author name and increase the color and thickness of the column-rule. Add an image and float the text around the image.
- 5.13** (*FBLM*) Modify the example in Fig. 5.16 to use a vertical flexbox.
- 5.14** (*Transformation with :hover*) Create a transformation program that includes four images. When the user hovers over an image, the size of the image should increase by 20%.
- 5.15** (*Reflection*) Create a reflection of an image 20px to the right of the original image.
- 5.16** (*Media Queries*) Create your own multicolumn web page and use media queries to adjust the formatting to use one column for mobile devices that have a maximum width of 480px.