

Viva La Vanilla Ice Cream

Read about iP-lhona Oil Paintings

CHAPTER 14

KNOW THE CODE

THE EXERCISES IN this chapter will provide technical lessons to help you understand and modify HTML and CSS code in Adobe Dreamweaver. You'll insert your content into the original template design. You should know that Dreamweaver is considered a WYSIWYG (What You See Is What You Get, pronounced "wizy wig") editor. As such, this software was created to help nonprogrammers "drag and drop" content into a page that is then automatically rendered in code. It's fine for a beginner to work in this manner. But over time, it becomes obvious to anyone using the software that it's more efficient (and successful) to alter the code directly rather than relying on the Dreamweaver tools and panels. In the exercises, I'll demonstrate ways of working with both Dreamweaver tools and code.

HTML: THE CONTENT GATEKEEPER

As discussed in Chapter 13, *Graphics on the Web*, HTML is an acronym for hypertext markup language. The term hypertext was developed by Ted Nelson in the 1960s to refer to “forms of writing which branch or perform on request; they are best presented on computer display screens” [1]. Nelson’s original view of hypertext was more generalized and complex than what we think of today as the “link;” however, his work provided the basis for linking on the web. Hypertext, hypermedia, and interactive media design are all related by what may seem today like the banal phenomenon of the hyperlink. While the invention of the link may seem underwhelming, the ability to link to files via the internet has revolutionized commercial communication, the interpretation of copyright laws, and social movements in both the private and public sectors. One of the primary tasks of the HTML file is to link one page to the next—and the images, a video, or other media files that you might see on the page are actually displayed via links to those files’ locations.

Olia Lialina’s *My Boyfriend Came Back from the War* (often referred to now as MBCBFTW) was one of the first net.art works to create a hyperlink narrative in 1996 using frames (FIGURE 14.1). Christiane Paul explains:

Links make it possible to connect texts and visuals to the contextual network in which they are embedded, and to visualize a network of references that would normally be separated by physical space. Within this network, information is subject to infinite recycling and reproduction, two concepts that form the basis of a multitude of online art projects [2].

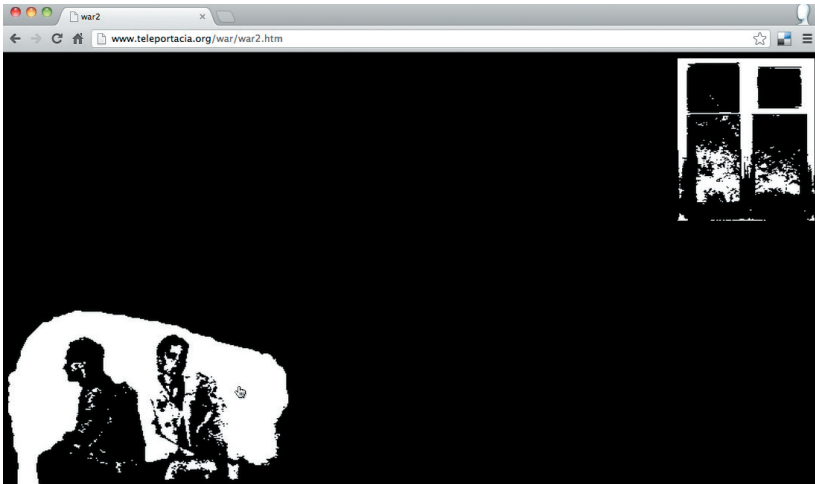


FIGURE 14.1 Screenshot of Olia Lialina’s *My Boyfriend Came Back from the War*, 1996.

Reference [1] Theodor H. Nelson, “Computer Lib/Dream Machines.” In *The New Media Reader*, ed. Noah Wardrip-Fruin and Nick Montfort (Cambridge, MA: The MIT Press, 2003), pg. 314.

Reference [2] Christiane Paul, *Digital Art* (London: Thames & Hudson, 2003), pg. 116.

Lialina’s early work has become a common point of departure for remixed versions of the original. One of my favorites is the recent M.B.C.B.F.T.W. (*Redux, At Rest*) by Freya Birren (FIGURE 14.2). Birren’s low-fi video reinterprets frames in the original net.art project as a series of sticky notes with a war story updated for 2012.

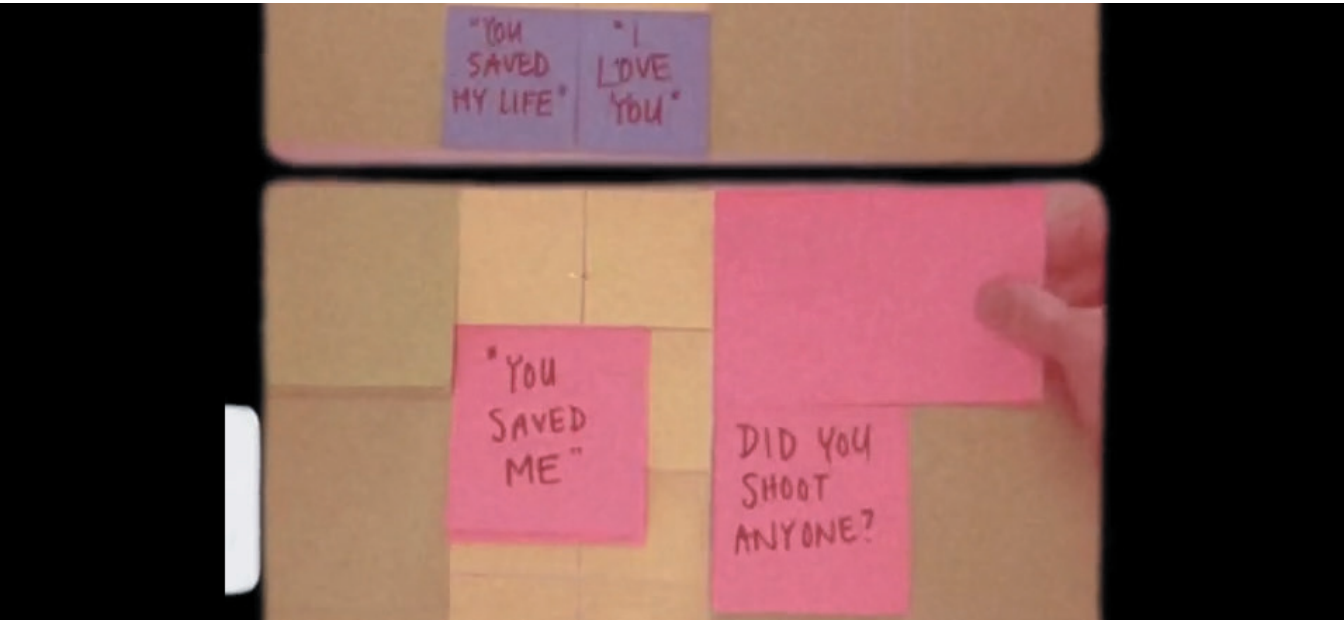


FIGURE 14.2 You can view Freya Birren’s 2012 low-fi version of Olia Lialina’s M.B.C.B.F.T.W. (*Redux, At Rest*) on Vimeo at <https://vimeo.com/44923030>.

You’re probably already familiar with using markup languages, even if you’ve never written a line of code. If you’ve structured a Microsoft Word or Open Office document with a centered title, perhaps, or a bold heading, you’ve used a markup language. These word processing applications provide the user with one-click access to selecting and modifying text (for example, transform a line of body copy into a greater hierarchical element such as a heading). Once text has been modified on the screen, it has also been modified in its file via the markup code. When the page of text is printed, the printer performs duties outlined in the markup code saved with the file. If you’ve printed your word processing file to confirm that the heading you created appears large and bold on a sheet of paper, you’ve essentially performed the same task as formatting content in the HTML file and previewing your work in a browser. The markup (applying a heading) applies structure to the document—the second task of the HTML.

There’s a difference between using markup to format or style content and using markup to apply structure to a file. At one time, HTML was used to control the presentation of content, but that’s no longer considered to be a best practice. That’s the role of Cascading Style Sheets (CSS).

`LINK ME UP`

The anchor tag, `<a>`, is used to create a hyperlink. It can be used to link one part of a page to another part of the same page. Or it can be used to link one document to another, to files saved on a server, to any destination on the web, or even to an email address.

In its simplest form, an HTML tag is a set of angled brackets enclosing the code for the element. Many HTML elements have two tags that surround a piece of content: an opening tag, `<a>`, and a closing tag, ``. The content that the anchor element surrounds becomes *clickable*—the set of words, or the image, becomes a link.

The anchor simply says to provide a link from the current document; obviously, it needs additional information. This is provided using HTML attributes that appear within the opening tag separated by a space. In the `<a>` tag, you'll almost always see the attribute `href` with a value in quotes such as: `href="http://peachpit.com"` telling the browser what to link to. In the code provided by the template, you'll see that the `href` attribute allows the anchor to point to a *hypertext reference*.

The title of this sidebar provides a simple example. That code, when rendered by a browser, would simply present the words "Link Me Up." The `<a>` tag makes the text clickable. The `href` attribute points to a (fake) document titled `mypage.html`. The `a` tag is then closed, ``, to prevent additional page content from linking.

HTML SYNTAX

HTML code always begins with the DOCTYPE and HTML tags to introduce the format of the document to the browser. This is followed by two major sections: head and body. The head area of the page includes what's referred to as meta information, that is, information about the web page, such as its title. It may also include any scripts that are attached to the page and links to external style sheets (FIGURE 14.3). The body of the page is where you'll spend most of your time. It includes all of the content that appears in the web browser. This content is surrounded by tags formatted with angled brackets (FIGURE 14.4).

FIGURE 14.3 The head section in HTML code, seen here in Dreamweaver's code editing view.

```
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
5 <title>HTML Structure</title>
6 <link href="oneColFixCtr.css" rel="stylesheet" type="text/css" />
7 </head>

9 <body>
10 <div class="container">
11 <div class="content">
12 <h1>This is a first level heading</h1>
13 <p>Place body copy between paragraph tags. Add links with the <a href=
"http://www.webmonkey.com/2010/02/html_cheatsheet/#Links">a tag and the href attribute</a>.</p>
```

FIGURE 14.4 The body tag and elements on the page in Dreamweaver's code editing view.

The tags are usually opened and closed surrounding content on the page. To create page structure, a tag is typically opened; then all of the content belonging to that tag is added to the page; then the tag is closed. For instance, after the head tag is closed using `</head>`, the `<body>` tag opens. At the end of the page, once all of the content has been inserted, the body tag is closed and directly following this, the HTML tag is closed using `</body></html>`.

SPACES IN THE CODE

HTML code will ignore the space bar when it's used more than once to add content to the page. These two sets of paragraph tags result in the same preview in the browser, despite the manually entered spaces:

```
<p>Here is some fine text.</p>

and

<p>   Here   is    some  fine   text   </p>
```

CSS: INSERT YOUR STYLE HERE

When it was first developed, all of the content that was placed in the HTML file was also formatted or styled using the same markup language. The development of HTML, however, was informed less by aesthetic considerations and more by structure and function. This has changed over time, as visual thinkers came to find HTML markup stifling to web aesthetics. Unfortunately, many of the techniques used to improve presentation undermined the content-structuring role of HTML.

Robert Cailliau, one of Tim Berners-Lee's colleagues at CERN who helped to develop the World Wide Web, was interested in separating content from presentation in the early stages of development. A new language was invented and slowly adopted by companies that developed web browsers. This new approach handled the formatting aspect of designing for the web and preserved HTML for structure and function. Enter CSS code.

CSS stands for *Cascading Style Sheets*. While HTML is used to apply structure to the content of a web page, CSS is referenced within the HTML file in order to apply formatting to the presentation of the structured content. CSS can be included directly in the HTML file (embedded CSS code appears in the head section of the HTML document and inline CSS styles appear within the body section of the code as part of a single tag), or it can be linked externally. Since externally linked CSS files can be applied to an entire website, most templates

O BROWSER, MY BROWSER

In 2012, I wanted to incorporate new elements in HTML5 (the fifth major revision of HTML) and CSS3 (the third revision of CSS) into a series of poems designed for reading in the web browser. I selected poems that represented the beginning, middle, and end of the 20th century: Walt Whitman's "O Captain, My Captain" from *Leaves of Grass* (1900), Jack Kerouac's novel/prose/marathon text *On the Road* (1957), and Adrienne Rich's "Waiting for You at the Mystery Spot" in *Fox* (2000). In each, the word "browser" is substituted for one of the words in the poem's title, forcing the poet to speak to the 21st-century web browser.

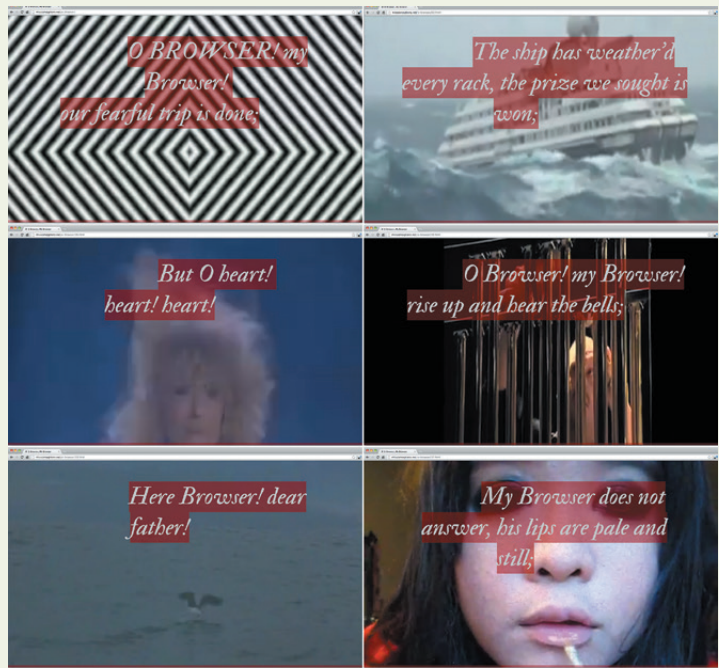


FIGURE 14.5 xtine burrough, *O Browser, My Browser*, 2012. <http://missconceptions.net/o-browser>. This project was created with the help of TERMINAL, an annual award for net art from Austin Peay State University. View the project source code for a list of video credits.

In all three browser poems, CSS is responsible for the aesthetics of the work. *O Browser, My Browser* (FIGURE 14.5) features background videos selected from YouTube (via the new HTML5 `<video>` element) and overlaid text formatted with CSS. The `@font-face` CSS property (which became fully available around the implementation of CSS3) is responsible for my use of uploaded typefaces on the website. In the original poem, the death of a ship's captain is an allegorical reference to the death of Abraham Lincoln. In this reinterpretation, the allegory shifts to the impending death of the web. Here, short clips of YouTube videos (all 24 found by a search on the site using keywords from each line in the poem) provide background noise that acts as a collage with the poem's text.

will provide a linked CSS file (typically called *styles.css*) referred to in the head section of the HTML page.

CSS SYNTAX

CSS uses a different syntax than HTML—rather than tagging content, a style sheet is simply a list of style declarations (called *declaration blocks*) that provide properties and values for selected parts of the HTML code. Instead of angled brackets, you'll notice curly brackets, colons, and semicolons. Remember, a style sheet is always used in conjunction with HTML code. In the most simplified form, a declaration block in CSS code, or rule, is formatted as follows:

```
Selector {  
Property: value;  
}
```

There are, of course, variations and complexities beyond the scope of this chapter. For instance, you might want to apply the formatting only in specific situations, such as when an HTML element is contained within a particular parent element.

GRAPHICS IN HTML AND CSS

As mentioned in the introduction to Chapter 13, a graphic file can be referenced in either the HTML or the CSS code. To determine where a graphic should be inserted, ask yourself the question: *Is this content or style?* A logo that's clickable (linked) to the home page is content. A background image over which you want to print type or links is considered part of the page's style. In Chapter 13, you inserted the updated logo graphic into the HTML code because it's considered to be part of the page content. In this chapter, you'll replace the background image of the firefighter with the soda jerk—a modification of the featured area's style dictated by the CSS code.

WHAT YOU’LL NEED

Download the following source materials to complete the exercises in this chapter:

- ✓ **chapter14-start** folder from the Chapter 14 downloads area on the companion website
- ✓ This is a duplicate of the files created in Chapter 13 with the animated gif, **bg-featured-home-soda.gif** saved in the images folder. You can learn to make this file by watching the Screencast 13-1.

You’ll benefit from the ability to be precise and watch out for typos during the modification process.

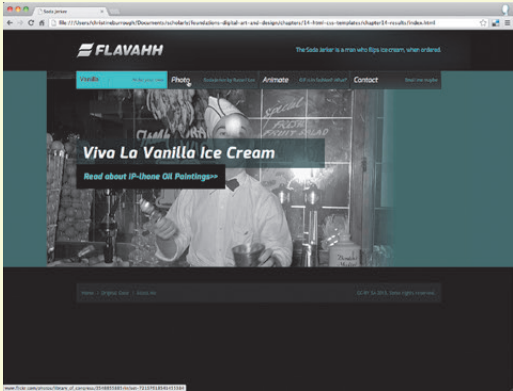


FIGURE 14.6 Completing exercises in this chapter yields a modified web page that integrates CSS with HTML code.

WHAT YOU’LL MAKE

In the exercises in this chapter, you’ll create a modified web page featuring the soda jerk photograph as the background image in the CSS of the featured content area, as well as updates to the content in the HTML file (FIGURE 14.6). In these exercises, you’ll be working in Dreamweaver.

EXERCISE 1 ANALYZE HTML IN DREAMWEAVER

Before making any modifications, set up your files and workspace and review the code provided by the template.

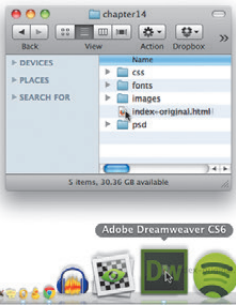


FIGURE 14.7 Mac users can open Dreamweaver by dragging a file on top of its application icon on the dock.

1. Copy the files you created in Chapter 13 to a new folder named **chapter14** and save the folder on your Desktop. (Alternatively, download the **chapter14-start** folder from the companion website, place it on your Desktop, and rename it **chapter14**.)
2. Rename file **index.html** to **index-original.html**. If you’re using a Mac, drag and drop **index-original.html** over the Dreamweaver icon in your dock (FIGURE 14.7). Alternatively, open Dreamweaver and choose the File menu > Open. Double-click **index-original.html** in the **chapter14** folder.
3. In Dreamweaver, choose the Window menu > Workspace Layout > Designer or select Designer mode from the pull-down list on the right side of the Application bar.

4. Set your view to Split by pressing the Split button on the top-right side of the Application bar or by choosing the View menu > Code and Design.
5. If the line numbers are not apparent in the code view, press the Line Numbers icon to the left of the code (FIGURE 14.8).
6. Analyze the code. Sometimes the easiest way to learn a new programming language is to review the code. Locate the following important sections of the HTML page:
 - A. **Line 1:** The DOCTYPE tells the browser that this is an HTML document (FIGURE 14.9).

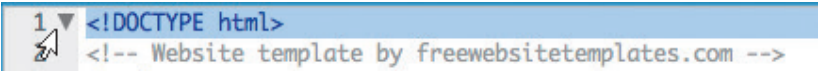


FIGURE 14.9 The Doctype element.

- B. **Line 3:** The `<html>` tag indicates the opening of the HTML code in the document. Notice that on Line 63 (the very last line of code), this tag is closed, `</html>` (FIGURE 14.10).
- C. **Line 4:** The `<head>` tag opens the head section of the HTML document. Meta information, such as the title of the page, is saved in this section. Links to other documents, such as CSS, JavaScript, or jQuery files, are also stored here. Notice that `</head>` appears on line 8, closing this section of the document (FIGURE 14.11).



FIGURE 14.11 The head section of the HTML code.

- D. **Line 9:** Immediately following the closing of the head section of the document, the body section is opened with the body tag, `<body>`. Remember, HTML is the content gatekeeper. All of the content displayed on the page will appear in this document between Lines 9 and 62, where the body tag is closed, `</body>` (FIGURE 14.12).

EXERCISE 2 MODIFY HTML CONTENT

You’ll begin this exercise by renaming the file from **index-original.html** to **index.html**. This will accomplish two important tasks: first, it creates a backup copy of the original file (named appropriately, so you can always return to the template if things go haywire). Secondly, web standards include the browser

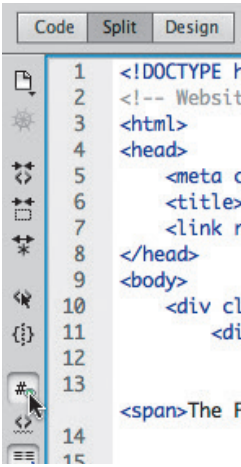


FIGURE 14.8 Display (or hide) line numbers in the code view.

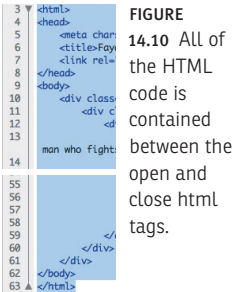


FIGURE 14.10 All of the HTML code is contained between the open and close html tags.

While these four elements are essential to any HTML document, you can rest easy knowing that when Dreamweaver is used to create a new HTML file, it automatically adds each of these elements to the code.

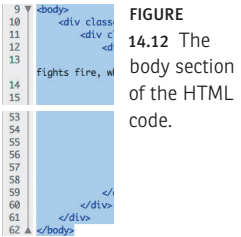


FIGURE 14.12 The body section of the HTML code.

automatically opening a page named “index,” which is stored in the top-level directory (the *root directory*) of any folder. For this reason, any page you want to save as the home page should be named **index.html**.

After saving the page with a new name, you’ll modify the HTML code in the head and body sections. You can modify code in Dreamweaver using Dreamweaver menus, panels, and icons or by directly manipulating the code. You’ll experience both practices and be mindful of how one side of the page view affects the other.

1. Choose the File menu > Save As and name the file **index.html**. Save it in the root directory for this project, which is the **chapter14** folder.
2. Notice that when the file is saved with a new name, the new (copy of the original) file opens in a new tab. Click on the X next to **index-original.html** to close the original file (FIGURE 14.13).
3. Your modifications will take place in **index.html**. Start by changing the title of the page, which will happen in the head area of the code. Change “Fayafayta Website Template” to “Soda Jerker” in the Title field within the Document Toolbar (FIGURE 14.14). Press the Enter key on your keypad to set the Title, or click anywhere in the design frame.

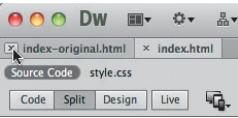


FIGURE 14.13 Dreamweaver files open in tabs. They can be closed by pressing the X on the tab.

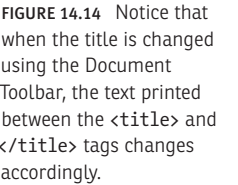


FIGURE 14.14 Notice that when the title is changed using the Document Toolbar, the text printed between the <title> and </title> tags changes accordingly.

The title is an important part of the HTML document because it provides a label for the page in the browser and the default text that’s saved when a user creates a bookmark. It’s also the single most important factor noticed by search engine “bots.”

4. Now you’ll begin modifying elements on the page, which all appear in the body section of the HTML page. Change the text, “The Firefighter is a man who fights fire, when ready” to “The Soda Jerker is a man who flips ice cream, when ordered.” This seems like it would be a fairly simple modification, but there’s a way to get into trouble while editing text in Dreamweaver. The easiest way to make this modification, in my opinion, is to edit the text within the code. If you prefer to work in the design frame, select only the words that you want to change (select “Firefighter” and replace it with “Soda Jerker”) and then move on to the next word or words you want to alter. To modify the text within the code, find the text between the and tags on **Line 13** and replace it.
5. The horizontal navigation tool is the next visual item on the page. It’s also the next part of the code in the body section. A traditional horizontal or vertical navigation tool will be coded as an unordered list in the HTML file

THE PERILS OF SELECTING IN DESIGN VIEW

You might be tempted to double-click on a word in design view in order to select it. That will work, so it may lead you to believe that clicking three times on a word will select an entire sentence that you intend to replace. In design view, this will also appear to be successful. However, as you can see in the code, when I triple-click on the word “Soda”—more than just the text is selected (FIGURE 14.15). If I replaced the sentence, I’d lose the Logo image and the link back to the home page (**index.html**) that the logo provides. *Yikes!* Dreamweaver often behaves in ways that are not entirely predictable. Always keep an eye on the code as you’re making selections and working in design view.

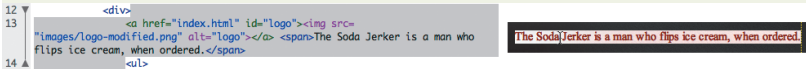


FIGURE 14.15 When I triple-clicked on a single word in the sentence in design view (on the right side of the application frame), all of the code on **line 13** was selected, including the image and the a tag. If I replaced the sentence, the image and a tag would be omitted, greatly changing the page layout and function.

with the tag. (It will also be styled in the CSS code, which you’ll see in the next exercise). The unordered list in your template file includes the following tags:

```
<ul>
<li>
<a href = "name of the page where the link goes">Clickable Link
Text</a><span>Some text about the link to the right of it</span>
</li>
```

...as many sets of and are listed as are needed for the full navigational tool...

```
</ul>
```

Change the clickable link texts and the text about the links in the code as follows (FIGURE 14.16):

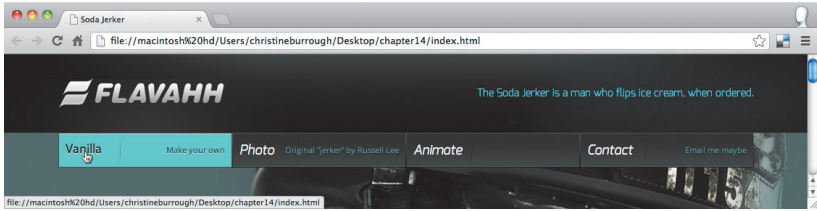
- A. Change “Home” to “Vanilla” and set the text about the Vanilla link to “Make your own.”
- B. Change “About” to “Photo” and set the text about the Photo link to “Soda Jerker by Russell Lee.”
- C. Change “News” to “Animate” and set the text about the Animate link to “GIF is in fashion? *What?*”
- D. Leave the “Contact” link as it is and set the text about the Contact link to “Email me maybe.”

```
14      <ul>
15      <li class="selected">
16          <a href="index.html">Vanilla</a> <span>Make your own</span>
17      </li>
18      <li>
19          <a href="about.html">Photo</a> <span>Soda Jerker by Russell Lee</span>
20      </li>
21      <li>
22          <a href="news.html">Animate</a> <span>GIF is in fashion again? What?</span>
23      </li>
24      <li>
25          <a href="contact.html">Contact</a> <span>Email me maybe</span>
26      </li>
27  </ul>
```

FIGURE 14.16 The links are modified between each of the list tags.

6. Press **Command(⌘)-S** to save the file and then preview it in a browser by choosing the **File** menu > **Preview in Browser** (select any browser). If you hover your mouse over the links and look toward the bottom of the browser window, you'll see the path that the hyperlink is set to load upon clicking the mouse (FIGURE 14.17). The first link, "Vanilla," loads the home page, `index.html`. However, the text, "Make your own," suggests that you intend to go somewhere else. In the next step, you'll change the settings (the `href` attribute of the `a` tag) for each link in the horizontal navigation tool.

FIGURE 14.17 Check your links in a browser.



7. Change the following hyperlinks by placing the cursor in each of the linked text boxes in design view and modifying the Link field in the Dreamweaver Properties panel at the bottom of the Application Frame:

- A. Click once anywhere in the word "Vanilla" in design view to see a blinking text cursor. View the Link field in the Properties panel. There you'll see that the link is set to `index.html`. Browse the web for a recipe for vanilla ice cream. When you find one that you like, copy the entire link (including the first part, `http://www.`) and paste it into the Link field. Press **Return** or **Enter** on your keypad to exit the Link field and view the code (FIGURE 14.18). Save your work and view it in a browser. The word Vanilla should link to a website with a recipe featuring vanilla ice cream. If it doesn't work, check your code!

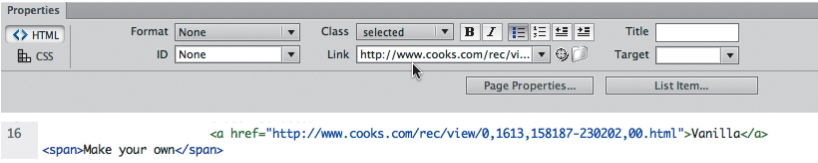


FIGURE 14.18 Use the Link field in the Properties menu to add or modify a hyperlink.

- B. Link "Photo" to the URL of the original photograph on the Library of Congress website or the Library of Congress Flickr photostream. I set the Photo link to the URL "`http://www.flickr.com/photos/library_of_congress/3548855885/in/set-72157618541455384`".
- C. Link "Animate" to a page that you'll create later, rather than a URL. Simply change the name of `news.html` to `animate.html` in the Link field or in the code.
- D. Finally, set the Contact link to open an email application and address an email to yourself. Instead of setting the `href` attribute to a page (`contact.html`), enter `mailto:youremailaddress@wherever.edu`.
8. Check the code. You should see a pattern where the `<a href` portion of the tag is green, the file or URL where the link goes is blue, and the angled bracket used to close the tag is green. Text that appears on the page (clickable or not) is black (FIGURE 14.19).

```
14      <ul>
15      <li class="selected">
16          <a href="http://www.cooks.com/rec/view/0,1613,158187-230202,00.html">Vanilla</a> <span>Make your own</span>
17      </li>
18      <li>
19          <a href="http://www.flickr.com/photos/library_of_congress/3548855885/in/set-72157618541455384">Photo</a> <span>
20      Original "jerker" by Russell Lee</span>
21      </li>
22      <li>
23          <a href="animate.html">Animate</a> <span>GIF is in fashion again? What?</span>
24      </li>
25      <li>
26          <a href="contact.html">Contact</a> <span>Email me maybe</span>
27      </li>
28  </ul>
```

FIGURE 14.19 Code is color-coated, and text that appears on the web page is black.

9. Select only the word "What?" in the text next to the Animate link and press the italic icon in the Properties panel. View the code. This word is surrounded by `` and ``, the emphasis and close emphasis tags (FIGURE 14.20).
10. Make the following simple modifications, which are similar to those you've already made:
- A. Change the featured dummy text "Lorem Ipsum Dolor Sit Amet" to "Viva La Vanilla Ice Cream."

The Properties panel serves the HTML and CSS files. Make sure that the HTML button on the far left of the panel is active when you want to modify the HTML code.

BE CAREFUL! I often see students copy part of a URL and paste it to the Link field, resulting in a nonfunctioning link. Typos are also easy to make. Look carefully at the code—one quote should follow the equal sign after the `href` attribute, then the complete URL, and finally a closing quote. The URL should appear in blue, while the rest of the tag is green.

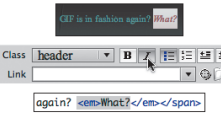
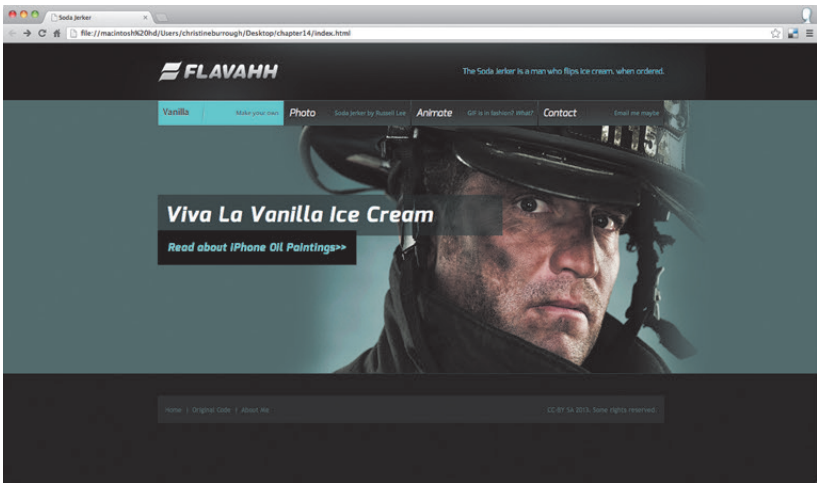


FIGURE 14.20 Top: Design view, where emphasis has been placed on the word "What?" Middle: The Properties panel, where emphasis can be added to the selected text. Bottom: Code view with emphasis tags surrounding the word "What?"

PLACEHOLDER LINKS You can use the pound sign (#) as a hyperlink reference that essentially goes nowhere. This is often used as a placeholder in the code (as Lorem Ipsum is used as dummy text) to indicate that a link will be created or inserted at a later date.

- B. Delete the second paragraph (“Vivamus dapibus fermentu elit.”)—do this in the code and be sure to delete the <p> and </p> tags surrounding the text.
 - C. Change “Read More>>” to “Read about iPhone Oil Paintings>>”.
 - D. Set the “Read More>>” link to the Hyperallergic essay on iPhone Oil Paintings by Kyle Chayka at <http://hyperallergic.com/57045/iphone-oil-paintings-j-k-keller/>.
 - E. Change the bottom navigational links. Leave “Home” set to [index.html](#), change “About” to “Original Code” (set the link to the original template at “<http://www.freewebsitetemplates.com/preview/firefighterwebsite-template/>”) and Change “Blog” to “About Me” (leave the link set to “#”). Delete the “Contact” link.
 - F. Finally, alter the bottom copyright information between the paragraph and close paragraph tags, <p> and </p>, beneath the final close unordered list tag, . Mine is “CC-BY SA 2013. Some rights reserved.” You can remove the copyright symbol if you like. (In the code, it appears as ©.)
11. Save your work and preview it in the browser (FIGURE 14.21). Click on your links to see if they work properly. (Remember, the Animate and About Me pages have not yet been created.)

FIGURE 14.21 A view of the work in a browser.



EXERCISE 3 LINK THE CSS FILE TO THE HTML DOCUMENT

Once again, you’ll begin this exercise by renaming a file in order to keep a back-up copy of the original. The original CSS file is named **style.css**, and it’s saved in the **css** folder in your root directory. When you rename this file, you’ll see how the link to it, once disconnected, removes the effects of all of the styles from the content of the page. Keeping the styles separated from the content is important because it lets you modify an entire website in one document, rather than a single element on one page. Obviously, the link to the CSS file must be set precisely in order for the styles to take effect.

1. **Line 7** in the head section of the HTML code in Dreamweaver includes the link tag, <link>, which is used to tell the browser how the current document relates to one that’s saved externally. This tag creates a relationship between two separate documents, and it’s often used to connect the HTML code with a style sheet that modifies content in the HTML. Notice that in <link>, the href attribute points to the file, **style.css**, which is saved in a directory (or folder) named **css**. This part of the code within the link tag href="css/style.css" is essential to the web page, as it links the styles saved in **style.css** to the content saved in **index.html**.
2. Open the **chapter14** folder on your Desktop. Within that folder, expand the subfolder, **css**. Rename **style.css** to **style-original.css** (FIGURE 14.22).
3. Toggle back to Dreamweaver. *Whoa!* Did you notice that in the design view your page suddenly feels empty? Preview your work in a browser. This is what an HTML page often looks like when its style sheet has been disconnected (FIGURE 14.23). Remember that the code on **Line 7** attaches a style sheet named **style.css** to this HTML document. Since you just renamed the style sheet in your Desktop folder, there’s no longer a style sheet named **style.css**. With no style sheet, there are no styles.
4. Return to Dreamweaver. Choose the File menu > Open and open the document **style-original.css** from the **css** folder inside the **chapter14** folder on your Desktop. When it opens, choose the File menu > Save As to save the file as **style.css** in the same **css** folder. Close the two CSS files that are open

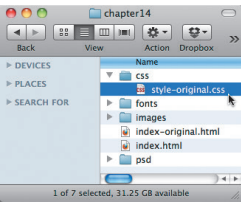


FIGURE 14.22 The style sheet is renamed.

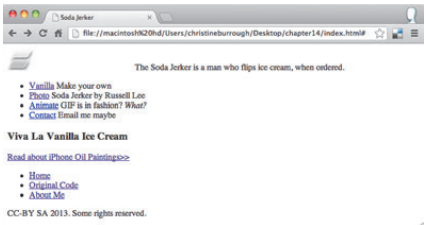
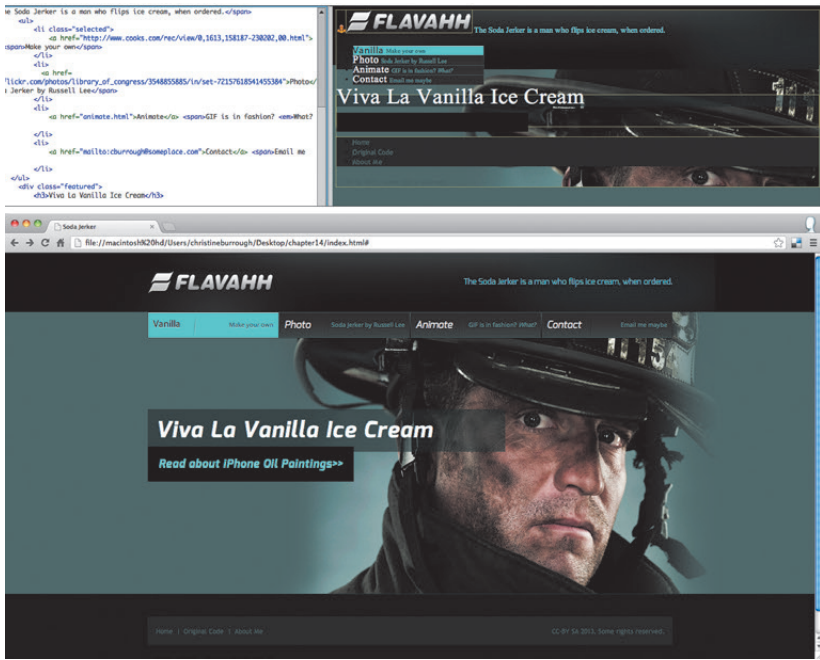


FIGURE 14.23 If the style sheet isn’t properly linked to the HTML file, the browser displays unstyled content.

When you see an HTML document that appears to have lost its styles, investigate the location and name set in the link tag href attribute. If you moved or renamed your style sheet, you’ll have to update <link> or save a new copy of the style sheet in the appropriate location.

and view `index.html` again. Since you’ve restored a file named `style.css` to the `css` folder, you’ll see that the styles are now reattached to the HTML document. However, as I was switching views, I ended up making my Dreamweaver Application Frame a bit small. Take a look at the differences between my design view in Dreamweaver and what the page actually looks like in a browser (FIGURE 14.24). You should always view your work in a browser, as Dreamweaver’s view can be unreliable.

FIGURE 14.24 Top: A view of the file in Dreamweaver. Bottom: A view of the file in the browser.



EXERCISE 4 ANALYZE CSS IN DREAMWEAVER

Now that you understand the relationship between the HTML and CSS documents, you’ll analyze the CSS syntax.

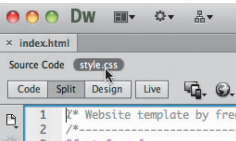


FIGURE 14.25 Access a linked style sheet by clicking its name in the application frame.

1. In the Document Toolbar, click on the word “`style.css`” to access the style sheet that’s linked to the `index.html` source code (FIGURE 14.25). Get familiar with clicking between the source code and style sheet, as you’ll often toggle between the HTML code to alter the content and the CSS code to modify the style sheet.
2. Scroll through `style.css`. This document may seem more repetitious than the HTML file: it’s a template that’s been well organized by the developer. This particular file is organized into sections that are printed using the CSS

commenting syntax. Comments are helpful because they enable the artist, designer, or developer to notate the code in plain language. **Line 1** includes the comment “Website template by freewebsitetemplates.com” surrounded by the commenting syntax slash asterisk... asterisk slash, as such:

```
/* Comments go in here */
```

3. View the comments that organize the document: styles that govern fonts begin after **Line 2**, layout styles begin after **Line 52**, styles for the header of the page begin after **Line 68**, styles for the body of the page begin after **Line 185**, and finally styles that control the footer begin after **Line 270**.
4. In each of these areas, you’ll see the same structure. CSS is written in concise declaration blocks that consist of a *selector*, followed by a curly bracket, then a set of *properties* and *values* (separated by a colon), and finally an ending semicolon. The declaration concludes with a closing curly bracket. Here’s an example from **Line 58** in the style sheet:

```
a {  
    outline:none;  
}
```

In this example, `a` is the selector. It refers to all `a` elements within the HTML code. The curly brackets surround the properties and values listed for all instances of `a` (that is, every time there’s a line of HTML code that begins with `<a`, as in ``, and so on). Here, there’s only one property listed: `outline`, and the value for that property is `none` (FIGURE 14.26). In summary, this simple CSS code sets the outline to none for all links on the page.

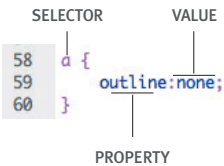


FIGURE 14.26 CSS code that modifies the `a` element in the linked HTML file.

5. Understanding CSS syntax is pretty simple. Locating where that CSS is being applied within the HTML code can be arduous. Advanced programmers can target particular parts of the HTML content by coding “parent and children” tags in the CSS document. For instance, one of the longer selectors in the CSS code provided in this template occurs on **Line 110**. The selector is as follows:

```
.header div ul li.selected span
```

The easiest way to understand where this is applied in the HTML code is to read it backwards. In other words, this selector targets a `span` tag, ``, that occurs within an `li` tag set to the class named “selected” that will look like this in the HTML document `<li class="selected">`. This occurs in a `ul` tag, ``, which occurs in a `div` tag, `<div>`, which occurs within a tag styled by the header class. Phew. To be more concise, the HTML code that takes the properties and values ascribed in the declaration

FIGURE 14.27 HTML code contains elements that are styled by a Cascading Style Sheet. On **Line 11**, the header class is assigned. A new division opens on **Line 12**. The unordered list tag is opened on **Line 14**. The first listed item appears on **Line 15**. Between Lines 16 and 17 the span tag surrounds the text “Make your own.”

```
11 <div class="header">
12 <div>
13 <a href="index.html" id="logo">
14 </a> <span>The Soda Jerker is a man who flips ice cream, when ordered.</span>
15 <ul>
16 <li class="selected">
17 <a href="http://www.cooks.com/rec/view/0,1613,158187-230202,00.html">
  Vanilla</a> <span>Make your own</span>
```

EXERCISE 5 SWAP A BACKGROUND IMAGE IN CSS

Once you understand where properties and values can be found in the CSS code, it’s not too difficult to modify the CSS file. You’ll change the background in the header area of the home page from a photo of a firefighter to Russell Lee’s soda jerker.

Just as with the HTML code, you can modify CSS code in Dreamweaver using Dreamweaver menus, panels, and icons or by directly manipulating the code. In this exercise, you’ll modify the style sheet by altering the CSS code directly. However, in the screencast for this chapter, I demonstrate how to modify the style sheet using Dreamweaver menus and panels.

1. To replace the firefighter photograph, you have to figure out where the image is placed within the code. The logo was inserted in the HTML code as content on the page. However, the firefighter image is set as a background image in the CSS code. If you try to click on the firefighter’s face in Dreamweaver, you’ll discover that his image cannot be selected. You can sort through code to find it; however, it’s easiest to find what you’re looking for when you know its name. Take a look in the **images** folder to find out the name of the firefighter image. Open the **chapter14** folder on your Desktop and then open the **images** folder. Fortunately, there aren’t too many images to view. By scrolling through the images, you’ll see that the image you’re looking for in the code is named **bg-featured-home.jpg**.
2. Since you know the name of the image, the easiest way to find it in the code is to execute a simple Find and Replace. With **style.css** open in Dreamweaver, choose the Edit menu > Find and Replace (or press **⌘-F**). Copy and paste the image name or type the name into the Find field exactly as it appears on your hard drive and press the Find Next button.

The file name will be highlighted in the code (FIGURE 14.28). Press the Close button once the code has been located.

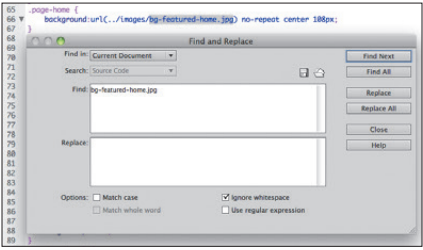


FIGURE 14.28 Use Find and Replace to modify code.

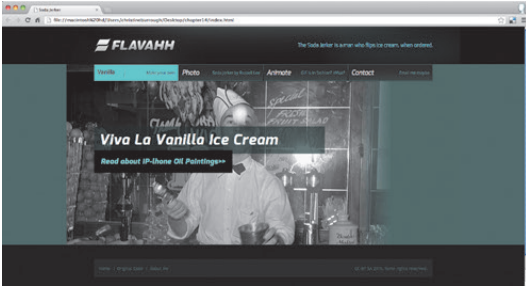
WATCH OUT! If you didn’t follow my naming conventions, my code adjustments won’t work in your files. Make sure that the name of the file you’re pointing to in your CSS code matches precisely the name of the file you saved on your hard drive.

PATH TO IMAGES

Make sure that your image is saved inside the images folder. The file path indicated inside the URL parenthesis includes two dots and a slash before the word images and then another slash before the file name: `../images/`. This formatting indicates that the soda jerker image is inside a folder named images that’s saved one level above the location where **style.css** is saved. (Check your hard drive—it’s true! You have to move out of the **css** folder to open the **images** folder, which is where the file is saved.)

3. In CSS, a background image is applied with the background or background-image property. Here the **bg-featured-home.jpg** image is set as the background image in the page-home class selector. In Chapter 13, you made the replacement image and saved it with the addendum **-soda** to the same name. Simply add this to the image name, so that the line of code setting the property and values looks like this:
`background:url(../images/bg-featured-home-soda.jpg) no-repeat center 108px;`
4. After you modify the name of the image, click anywhere in the design view of the Application Frame to see the image swap from the firefighter to the soda jerker.
5. Save your work and preview it in a browser (FIGURE 14.29).

FIGURE 14.29 Preview your work in a browser.



EXERCISE 6

CREATE AND LINK TO A NEW HTML PAGE

The Soda Jerker website is nearly complete. However, there are two links that currently go nowhere: the Animate button in the top navigational bar, and the About Me link in the footer. If you followed the screencast in Chapter 13, you created an animated GIF in which the ice cream passes from the scooper to the cup. Even if you didn't follow the screencast, you have my copy of the animated GIF in your **chapter14** folder, as I placed it in the **images** folder of the **chapter14-start** folder that you downloaded from the companion website at the start of these exercises. The new page will be identical to the home page, the only difference being the background image. Linking to new pages is simple, you need only modify the directions you give to the web browser following the href attribute of the a tag.

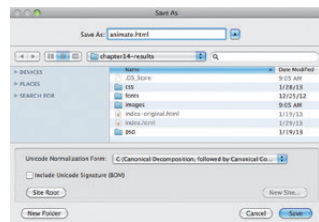


FIGURE 14.30 Save a new page in the root directory.

FIGURE 14.31 Notice the link to the background GIF. Do you see anything out of place?

1. Choose the File menu > Save As and save **index.html** as **animate.html** in the root directory (FIGURE 14.30). Notice that the new file is open in a new tab in Dreamweaver. The old file, **index.html**, is also open in a tab.
2. On the **animate.html** page, change the background image in the **page-home** class selector, just as you did in Exercise 5. Can you predict the problem you're going to face? If not, you'll see it by the end of Step 5! Click **style.css** for the **animate.html** page and replace the file named **bg-featured-home.jpg** with **bg-featured-home.gif** (FIGURE 14.31) in the **page-home** class. Essentially, you're changing the extension of the name.

```
65 .page-home {
66   background:url(../images/bg-featured-home-soda.gif) no-repeat center 108px;
67 }
68 /*----- Header -----*/
```

3. Choose the File menu > Save All to save both the HTML and CSS files.
4. Click on the **index.html** tab to work in the first document. Notice that the Animate button is already pointing to a page named **animate.html** (FIGURE 14.32). Preview your work in a browser.

FIGURE 14.32 The animate button references a page named animate.html.

```
21 </li>
22 <a href="animate.html">Animate</a> <span>GIF is in fashion? <em>What?</em></span>
23 </li>
```

5. *Doh!* While this is difficult to show in a static image, what you should have noticed is that the background animates on the home page and on the "animate" page (FIGURE 14.33). The style sheet, **style.css**, is referenced on both **index.html** and **animate.html**. So any change that you make to

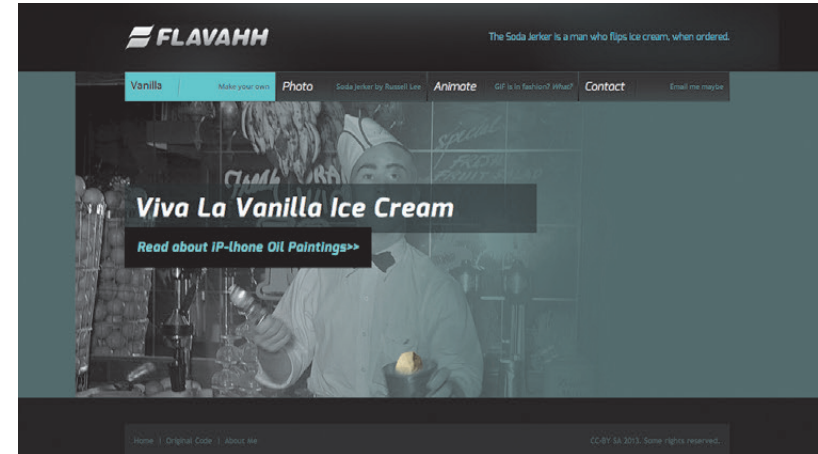


FIGURE 14.33 Notice that the position of the scoop of ice cream has moved when you view the work in a browser.

the style sheet is going to affect both pages (and, presumably, any other pages duplicated and saved to create the website). In the next steps, you'll remedy this situation by creating a separate class in the style sheet for the animation.

6. It doesn't matter if you work in **style.css** from **index.html** or from **animate.html**, since both HTML pages reference the same CSS file. Click to the style sheet from either open document.
7. Set the name of the **bg-featured-home.gif** image back to a JPG file, **bg-featured-home.jpg**, in the **page-home** class. (Undo what you did in Step 2.)
8. Copy the entire class and paste it to a new line in the CSS document. Be sure to start with the period before the class name and end with the closing curly bracket.
9. Change the name of the new class to **page-home-animate**; then change the extension of the background image to **.gif** (FIGURE 14.34).

```
65 .page-home {
66   background:url(../images/bg-featured-home-soda.jpg) no-repeat center 108px;
67 }
68 .page-home-animate {
69   background:url(../images/bg-featured-home-soda.gif) no-repeat center 108px;
70 }
```

FIGURE 14.34 Change the extension of the background image to .gif in the CSS.

10. Click the tab to work in the **animate.html** document. Then choose the source code button next to **style.css** to modify the HTML code. Extend the name of the class in the opening div tag by adding **-animate**. (In my document, this appears on **Line 10**; it will be the first tag that opens beneath the body tag.) In other words, the tag was **<div class="page page-home">** and becomes **<div class="page page-home-animate">** (FIGURE 14.35).

```
<body>
<div class="page page-home-animate">
  <div class="header">
```

FIGURE 14.35 Change the name of the div class in the HTML document to match the class you want to use in the CSS code.

11. Save your work and preview it in a browser. You should be able to click on the FLAVAHH logo to see the home page (FIGURE 14.36) and the Animate button to see the page with the animated background (FIGURE 14.37). Both pages use the same style sheet, and each calls its own background image into the featured section of the page.

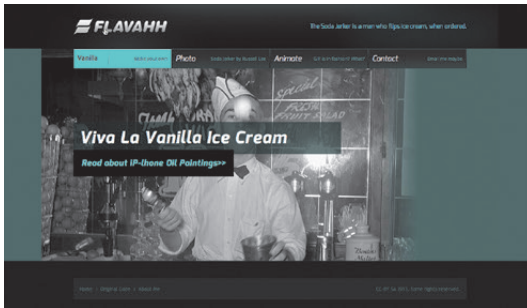


FIGURE 14.36 Click the logo to return to the index page.



FIGURE 14.37 Click the Animate button to see the page with the animated background image.

SCREENCAST 14-1 MODIFY CSS CODE USING DREAMWEAVER PANELS

In Exercise 5, you adjusted the CSS code directly in the style sheet. To see how to modify the CSS style sheet using Dreamweaver menus, panels, and icons, view this chapter's screencast.

All screencasts are available on the companion website, www.digitalart-design.com, or on the YouTube playlist, www.youtube.com/playlist?list=PLAy6P5IoEjy2v3kZKt8spqJ50nLb2XQl.

LAB CHALLENGE

Create a new page! To complete this page, you need to make a new page that opens when a visitor clicks on “About Me” in the footer set of links.