

Morten  
Rand-Hendriksen

24 Proven One-hour Lessons

Sams **Teach Yourself**

Microsoft®

# Expression Web 2

in **24**  
**Hours**



SAMS

## **Sams Teach Yourself Microsoft® Expression Web 2 in 24 Hours**

Copyright © 2009 by Pearson Education, Inc.

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher. No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

This material may be distributed only subject to the terms and conditions set forth in the Open Publication License, v1.0 or later (the latest version is presently available at <http://www.opencontent.org/openpub/>).

ISBN-13: 978-0-672-33029-2

ISBN-10: 0-672-33029-6

### *Library of Congress Cataloging-in-Publication Data:*

Rand-Hendriksen, Morten.

Sams teach yourself Microsoft Expression Web 2 in 24 hours / Morten

Rand-Hendriksen.

p. cm.

ISBN 978-0-672-33029-2

1. Web sites—Authoring programs. 2. Microsoft Expression Web. I.

Title. II. Title: Teach yourself Microsoft Expression Web 2 in 24 hours.

III. Title: Microsoft Expression Web 2 in 24 hours.

TK5105.8883.R36 2008

006.7'8—dc22

2008034613

Printed in the United States of America

First Printing September 2008

### **Trademarks**

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Sams Publishing cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

### **Warning and Disclaimer**

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an “as is” basis. The author and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book.

### **Bulk Sales**

Sams Publishing offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales. For more information, please contact

**U.S. Corporate and Government Sales**

**1-800-382-3419**

**corpsales@pearsontechgroup.com**

For sales outside of the U.S., please contact

**International Sales**

**international@pearsoned.com**

### **Associate Publisher**

*Greg Wiegand*

### **Acquisitions Editor**

*Loretta Yates*

### **Development Editors**

*Todd Brakke*

*Anne Goebel*

### **Managing Editor**

*Patrick Kanouse*

### **Project Editor**

*Mandie Frank*

### **Copy Editor**

*Mike Henry*

### **Indexer**

*Tim Wright*

### **Proofreader**

*Arle Writing  
and Editing*

### **Technical Editor**

*Greg Kettell*

### **Publishing Coordinator**

*Cindy Teeters*

### **Designer**

*Gary Adair*

### **Compositor**

*Mark Shirar*

# Introduction

## About This Book

Because you have opened this book and started reading the introduction, I am assuming that you are interested in learning how to create websites using Microsoft's new and exciting web development application, Expression Web 2. If so, I congratulate you: By choosing this application as your platform, you are already well on your way to creating functional and well-designed sites based on web standards. In other words, unlike me, you are putting the proverbial horse before the cart and starting at the beginning rather than learning things the hard way.

Expression Web 2 is the second version (duh!) of a web design and development platform that sees Microsoft take a whole new approach to the concept of web standards. As a result, you now have an application that produces standards based websites right out of the box without requiring any tweaking or custom coding on your end. In fact, using Expression Web 2, you can create advanced standards-based websites with lots of fancy interactive features without ever writing a line of code. And with that the threshold for learning, understanding, and creating websites that look and behave the same across all browsers and platforms is lowered to a level anyone can manage.

When I started writing this book, I spent a lot of time thinking about you, the reader; more specifically how to ensure that after reading this book, you would walk away with not only an understanding of the application but also how to use it to get from an idea to a finalized product. The result of my ponderings was a website called MyKipple.com that showcases many of the basic and more advanced functions available in Expression Web 2. By following the tutorials in this book, you will build the MyKipple.com website from scratch and in the process learn how all the different elements come together and how to get the most out of the application. When you have finished the last hour and the site is complete, you will have both the tools and know-how to build your own websites using HTML (Hypertext Markup Language), CSS (Cascading Style Sheets), and the many other functions that Expression Web 2 offers. You will also have a basic understanding of how the application deals with more advanced coding languages such as ASP.NET and PHP, and you'll even learn how to publish a simple Silverlight application.

Accompanying this book is a small web site that contains information about the book itself along with the lesson files for each hour and in time a Wiki or a forum for you the reader to get further information and showcase your work. The web site is located at <http://expression.pinkandyellow.com>.

## Who Is This Book For?

I always ask people why they do what they do because it says volumes about the end result. Therefore it is only fair that I ask myself the same question and provide you with the answer. So, Morten, why did you write this book?

When I started building websites in the 1990s, I really had no clue what I was doing other than a vision of what I wanted to create. And when I looked around for help in the form of books or tutorials, I couldn't find anything that spoke to me. What was available was either too technical or too pointless. No matter where I looked, I could only find basic algorithms on how to perform simple tasks, never detailed explanations of how to get from point A to point B. As a result, I ended up teaching myself how to do things. Needless to say I learned the true meaning of the term "taking the long way around."

So, when I signed up to write this book, I had one main goal in mind: To write the book I was looking for and really needed when I started out—a book that took me all the way from a basic sketch on a napkin to a fully working website and taught me how to use the application at the same time.

As a result, this book is written with the novice designer in mind. No, let me rephrase that: As a result, this book is written in a way that a novice designer will understand and learn from. I make this distinction because even if you are a seasoned designer or developer, I am certain you will find lots of useful information inside these covers.

Being introduced to Expression Web 2 has had an enormous impact on my business because it reduced what used to take hours or days to only minutes. In particular I am talking about the application's excellent CSS features. More than just a design and development application, Expression Web 2 is a tool that helps you learn, understand, dissect, and modify style sheets with unprecedented speed and accuracy. Getting a firm grasp on these tools and understanding how to use them in real-world scenarios will almost certainly make your work with CSS more efficient and productive, regardless of whether this is the first time you've encountered the term *style sheets* or you are a seasoned professional with years of experience.

So, without further ado, I wish you a pleasant journey and hope you come out on the other side with the skills, understanding, and confidence to take on the wild and exciting world of web design.

Lykke til!

Morten Rand-Hendriksen, July 14, 2008

## HOUR 4

# Building the Home Page— A Look Behind the Curtain

---

### ***What You'll Learn in This Hour:***

- ▶ How to read and edit basic HTML code in Code view
- ▶ How to import text from Microsoft Word and other documents
- ▶ How to clean up imported code using Code and Design views
- ▶ How to use Find and Replace to edit many instances of code at the same time

## **Introduction**

In Hour 3, “A Website Is Really Just Text—Build One in 5 Minutes,” you built a basic web page using functions similar to those in a word processing program. But that is just half the story: In web design, what happens behind the curtain is what really matters. It's time to learn some basic HTML and see how the code affects the content. To do this, you build the basic structure of the home page of the My Kipple site.

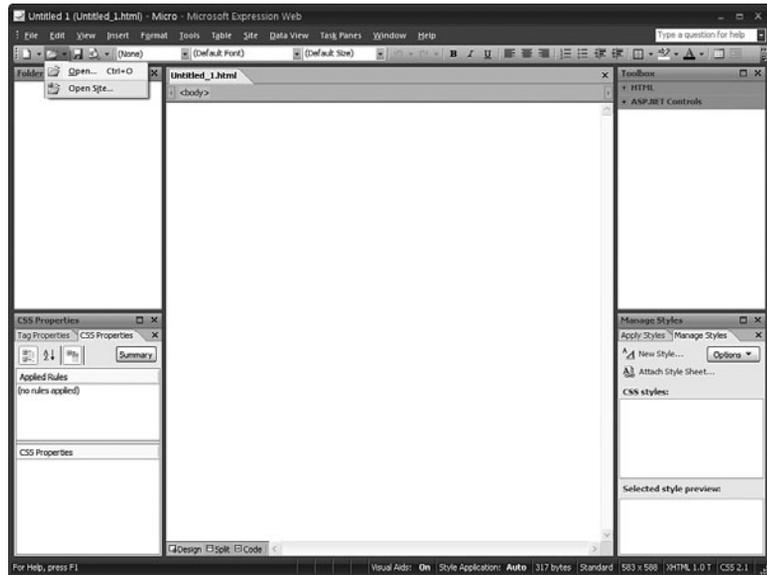
## **Opening and Editing an Existing File Using Code View**

If you haven't already done so, open the `default.html` file you worked on in Hour 3. The MyKippe site should open automatically when you open Expression Web 2. If not, you can find it by selecting Open Site from the File menu or from the drop-down menu of the Open icon on the Common toolbar (see Figure 4.1).

The Open Sites dialog lists all the sites created in Expression Web 2 (see Figure 4.2). When you create or define a website in Expression Web 2, the program automatically generates a shortcut to facilitate easy access to this list. If you can't find the project on this list, you can navigate to it as you normally would. If you followed the directions in Hour 3, the project is in the My Web Sites folder under My Documents.

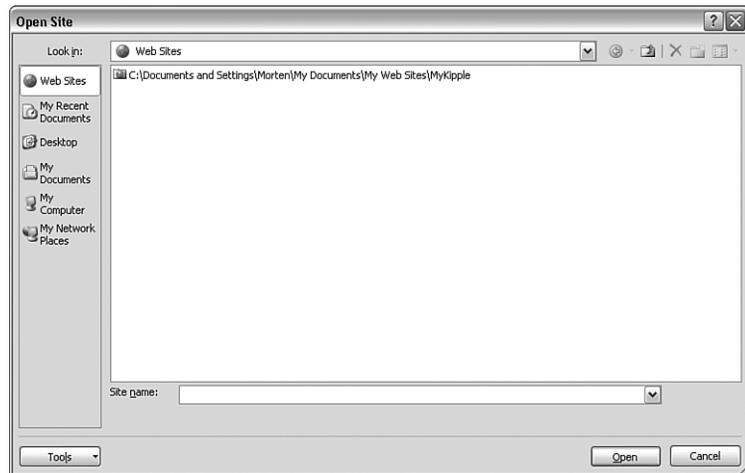
**FIGURE 4.1**

The drop-down menu under the Open icon on the Common toolbar gives you quick access to open files and sites without having to go to the Main menu



**FIGURE 4.2**

The Open Site dialog displays all the websites you have created in Expression Web 2. Here seen with the My Kipple project.

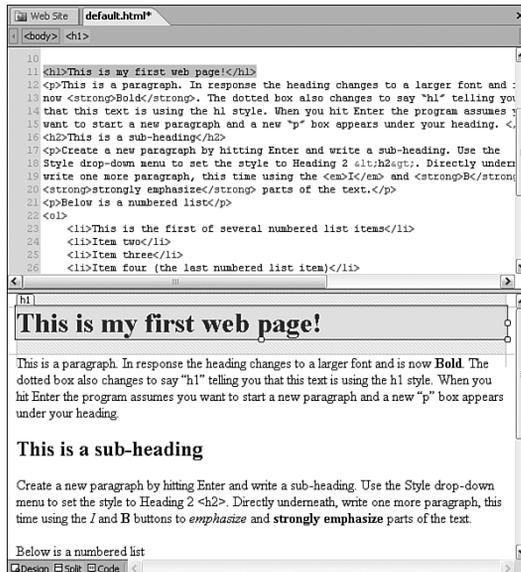


***Did you Know?***

The Open Sites dialog isn't restricted to local sites. You can also use it to open and manage external websites both through HTML (Hypertext Markup Language) and FTP (File Transfer Protocol) as long as you have the necessary access codes. In fact you can use Expression Web 2 to make changes to live websites in real time.

Before going any further, let's look at what is happening in the page's code. Select Split view using the button at the bottom of the pane to reveal both Code view and

Design view. Click anywhere on the heading and then click the h1 tag on the Block Selection box. This highlights the content affected by the h1 style in both Design view and Code view (see Figure 4.3). In Design view, you see a box with a gray striped-out



**FIGURE 4.3**

When clicking on an object in either Code or Design view that object is highlighted in both for easy reference.

area above and below. The gray areas are the default margins for the h1 style. In Code view, you see the text buffered or wrapped on both sides by code tags. These tags tell the browser to display the text in the h1 style.

HTML is a basic code language that can be summarized in one simple rule: Everything is wrapped between a beginning and an end tag. All beginning tags consist of a less-than bracket <, the tag name and/or function, and a greater-than bracket >. End tags look much the same, but with the addition of a forward slash / before the content. In this example, <h1> tags wrap the heading like this:

```
<h1>This is my first web page!</h1>
```

You now have two ways of changing the style of your content. You can use the style drop-down menu as you learned in Hour 3 or you can go into Code view and change the style manually. Try changing the <h1> and </h1> tags to <p> and </p>, and see what happens in Design view.

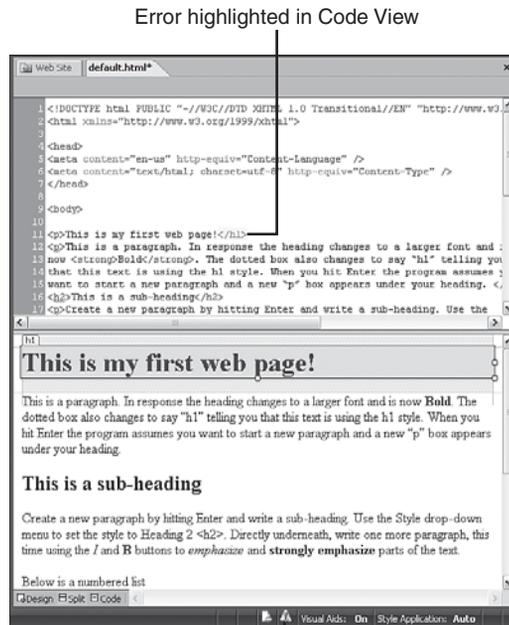
To use Design view to see changes you made in Code view, you have to click inside the Design view area. This is because as you change the code in Code view, you are temporarily breaking the code. Rather than trying to display broken code, Expression Web 2 waits for you to tell it when you finish editing before it refreshes the WYSIWYG (What You See Is What You Get) editor.

**Watch  
Out!**

Notice that when you change the beginning tag, Expression Web 2 highlights the end tag in yellow and red to tell you that your code is broken (see Figure 4.4). Likewise the

**FIGURE 4.4**

When Expression Web 2 discovers a code error it highlights the error with a yellow background and red text as well as showing a warning sign in the Status Bar.



Warning sign in Status Bar indicating error in Code View

status bar on the bottom of the workspace puts up two warning signs: The first one tells you that it detects an HTML incompatibility; the second one tells you that it detects a code error.

By studying the code, you see that all the different styles you applied in Hour 3 are actually the same type of tags:

- ▶ Heading 1 <h1>
- ▶ Heading 2 <h2>
- ▶ Paragraph <p>
- ▶ Italicized (emphasized) <em>

- ▶ Bold (strongly emphasized) `<strong>`
- ▶ Bullet (unordered) list `<ul>`
- ▶ Numbered (ordered) list `<ol>`
- ▶ List item for both lists `<li>`

The `</body>` tag, which wraps all the content, tells the browser that this is the content to display. The `<head>` tag contains all the meta information that is available to the browser but that the browser does not display within the page. Meta information includes the page title displayed at the top of the window, info about the designer, and so on. Finally the `<html>` tag, which tells the browser that the following content is written in the HTML language, wraps both the `<head>` and `<body>` sections. Scroll to the bottom of the Code view and you can see the `</body>` and `</html>` tags that close the page.

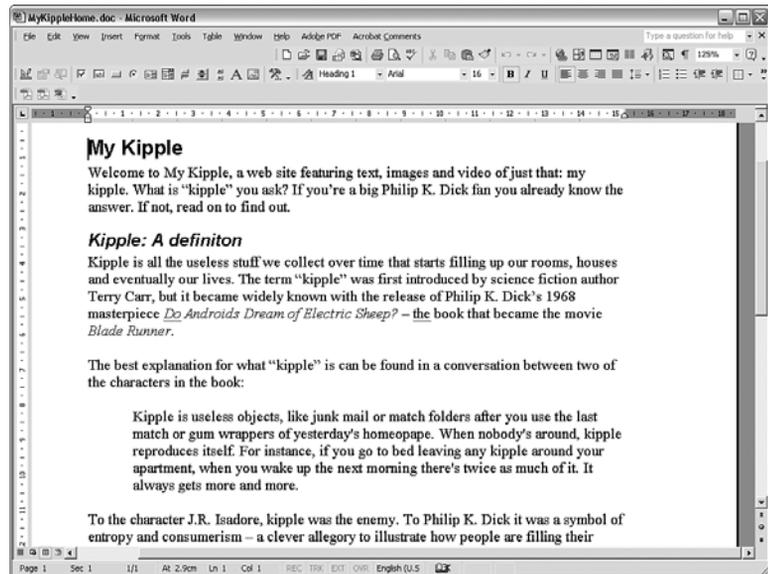
## Importing Styled Text from a Document

What is a website if not a source of information? You need some real content to work with. Right now the `default.html` file contains whatever text you inserted during Hour 3. The next step is to introduce some real content. In most cases, you will either be provided with or write your own content in some form of word processing software. Your first instinct is probably to cut and paste this content straight into your page. The problem is that word processors attach a large amount of invisible styling code to their documents—code that becomes very difficult to work with after import into an HTML file. To get rid of most of this superfluous code, Expression Web 2 can import this content and translate basic layout and styling for you. It's not a perfect solution, but it's far better than cutting and pasting.

In the set of files you downloaded from the book site is a Microsoft Word document called `MyKipp1eHome.doc` (see Figure 4.5). Open this document in Microsoft Word and you see a standard document with headings, subheadings, and some text.

The next step is to move all this content over to the `default.html` file and translate the styling to standards-based code. To import the content, you use the File command.

**FIGURE 4.5**  
**MyKippleHome**  
**.doc** as it ap-  
 pears in Mi-  
 crosoft Word.



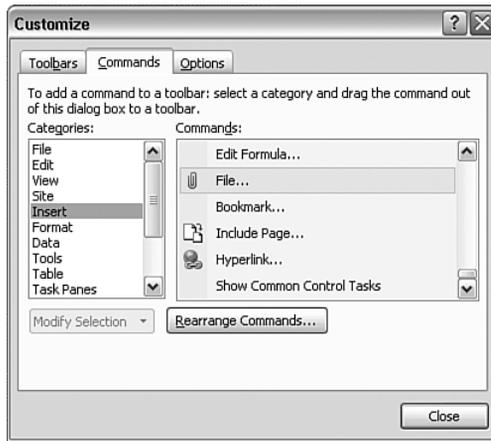
## ▼ Try it Yourself

### Add the Insert File Command to the Common Toolbar

Expression Web 2 is loaded with functions, many of which are not part of the standard toolbars. The Insert File command is one such function. But because you'll use it often in real life, add an Insert File command button to your toolbar.

1. Select Add or Remove Buttons and Customize under the Toolbar options as explained in Hour 1, "Get to Know Microsoft Expression Web 2." Alternatively, select Tools, Customize from the menu bar.
2. With the Commands tab selected, click Insert under Categories.
3. Scroll to the bottom of the Commands menu and find the File option with a paperclip icon next to it (see Figure 4.6).
4. Click the File option and drag it to the far right side of the Common toolbar. Release the mouse button and the paperclip appears on the toolbar next to the Font Color button (see Figure 4.7). Close the Customize dialog.

Before inserting new text, delete all the old content from the default.html file. In Design view, highlight all the content and press the Delete button on your keyboard. You should now have a blank page in Design view and the basic framework of an HTML file in Code view.



**FIGURE 4.6**  
From the Customize dialog you can add new functionality like the Insert File command to your toolbars.



**FIGURE 4.7**  
When you add new functions to the toolbars they will appear in the first open space to the right.

To import the content of the `MyKippleHome.doc` file into your project, click the Insert File button you just added to the Common toolbar. This opens a standard Select File dialog. Browse to the location where you saved the `MyKippleHome.doc` file. By default, the Insert File command looks for HTML files. But if you click the Files of Type drop-down menu, you see the program supports a long list of file types including Rich Text Format (.rtf), many different versions of Microsoft Word documents (.doc), Microsoft Excel worksheets (.xls), WordPerfect documents (.wpd), and many more. Simply clicking All Files (\*.\*) shows you all the available files. Select the `MyKippleHome.doc` file and click Open.

Depending on the version of Microsoft Word installed on your computer, a warning saying **Microsoft Expression Web 2 needs a converter to display this file correctly. This feature is not currently installed. Would you like to install it now?** might appear the first time you use the Insert File command. If you click Yes, Expression Web 2 asks you to insert the Microsoft Office CD and install some new components. If you don't have the Office CD handy, simply click No. Because of a quirk in the program, the function still works the way it should but you see the same warning every time you use the function until you install the necessary components.

**Watch  
Out!**

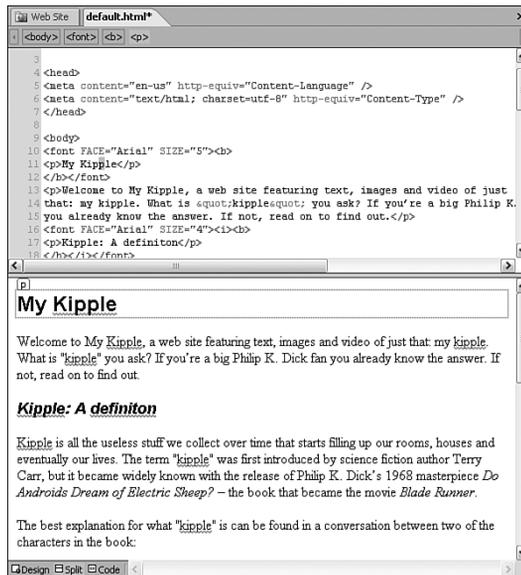
## Cleaning Up Imported Text

Although the new imported content looks the same in Design view as it did in Microsoft Word, the code tells a different story (see Figure 4.8). By highlighting the heading at the top of the document, you see that the code is no longer as clean as it was earlier:

```
<font FACE="Arial" SIZE="5"><b>
<p>My Kipple</p>
</b></font>
```

**FIGURE 4.8**

Although the new content looks right in Design View, the content imported from the word document is littered with “bad” code that is not standards based.



You can see that rather than setting the heading to the h1 heading style, the font type and font size are set with two commands: FACE and SIZE. Although this way of styling content works, it is both cumbersome and messy and creates a lot of extra work for anyone who wants to change the content later. For instance, if you have several headings like this throughout your document, you have to insert all that style code every time you add a new heading. As you will see when you start adding more advanced styling to your document, this way of defining the look of your text is also limiting. Furthermore the font element is deprecated meaning it is no longer supported by the W3C and should be avoided if at all possible.

You need to get rid of all this extra code and to clean up the new document. To do this, you can make use of both Design view and Code view. First attach the proper styles to the content. As you can see when you click the heading, Expression Web 2 defines it as a paragraph with some extra styling attached. To fix this, simply set the style to Heading 1 <h1> by using the Style menu as you did before. Use the same technique to set the three subheadings to h2. HTML has a dedicated tag, `blockquote`, for indented or highlighted paragraphs. Select the indented paragraph and use the Style menu to restyle the paragraph with Block Quote.

Now that all the sections have the proper tags, you can start deleting all the unnecessary code. In Code view, take away all the code before and after the heading tags. Afterward each heading should look like this:

```
<h1>My Kipple</h1>
```

Next you need to get rid of all the extra code attached to the indented section. Because the Block Quote style automatically indents the paragraph, all the other tags are now unnecessary. Right now you have this:

```
<dir>
  <dir>
    <span LANG="EN-CA">
      <blockquote>Kipple is useless objects, like junk mail or match folders
after you use the last match or gum wrappers of yesterday's homeopape. When
nobody's around, kipple reproduces itself. For instance, if you go to bed
leaving
any kipple around your apartment, when you wake up the next morning there's
twice as much of it. It always gets more and more.</blockquote>
    </span>
  </dir>
</dir>
```

With the superfluous `<dir>` and `<span>` tags deleted the section should now read

```
<blockquote>Kipple is useless objects, like junk mail or match folders after you
use the last match or gum wrappers of yesterday's homeopape. When
nobody's around, kipple reproduces itself. For instance, if you go
to bed leaving any kipple around your apartment, when you wake up
the next morning there's twice as much of it. It always gets more
and more.</blockquote>
```

Finally, change the italicized and bolded words to emphasized and strongly emphasized. You can either do this manually by replacing each individual tag in Code view or in Design view by highlighting each section and clicking its respective emphasis button twice—once to get rid of the old code and once to insert the new code. This document has only a few emphasized and strongly emphasized sections, so it's easy to make the changes manually. But because you often have to work with larger documents with many such instances, Expression Web 2 has a dedicated function to do all this work for you.

## Try it Yourself

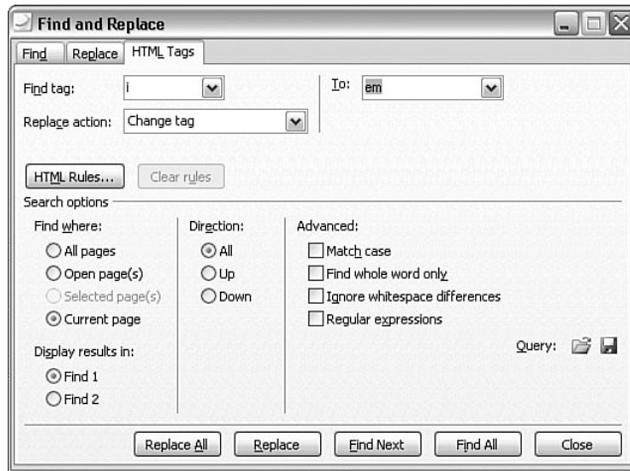
### Use Find and Replace to Change Multiple Tags at Once

Find and Replace is a great tool to find content and make multiple changes to a document with only a few mouse clicks.

1. Open the Find dialog by clicking Edit, Find on the menu bar or by using the shortcut Ctrl+F. Click the HTML Tags tab.
2. In the Find Tag area, enter `i` or select it from the drop-down menu. Under Replace Action, select Change Tag from the drop-down menu. Doing so opens a third option called To. Enter `em` or select it from the drop-down menu (see Figure 4.9).

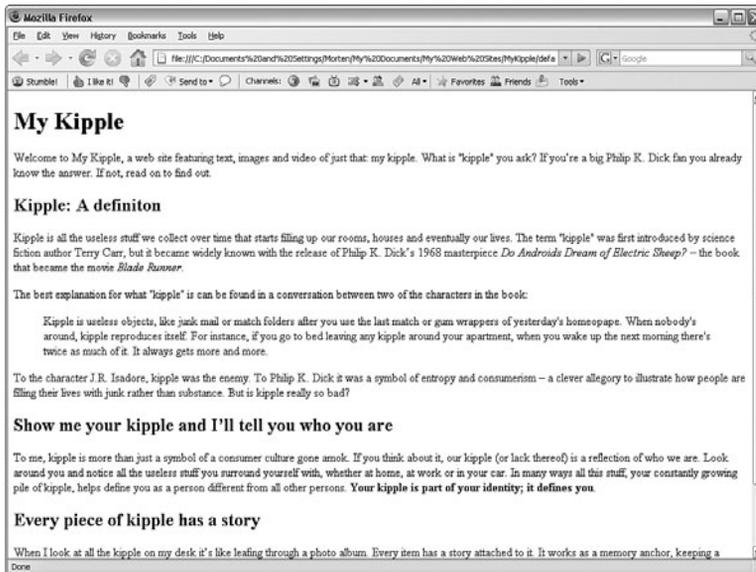
**FIGURE 4.9**

The HTML Tags tab under the Find and Replace dialog can be used to make document-wide changes to specific tags.



3. Click either Replace All if you are confident you want to replace all the tags, or click Find Next and then Replace for each instance. When you do so, the program replaces both the beginning and end tags throughout the document.

You have now successfully imported and converted the contents of a Word document to standards-based code (see Figure 4.10). The final step is to save the page on top of your old one (Ctrl+S) and test it in multiple browsers to ensure that it looks the way it's supposed to.



**FIGURE 4.10**  
The new **default.html** file as it appears in Firefox.

## Summary

In this hour you took a trip behind the scenes and learned a bit about how an HTML page works. You saw that the code boils down to wrapping the content in a `beginning` and an `end` tag and that changing these tags can be done from Code view and Design view and also by using Find and Replace. You also learned how to import text content from external files and clean up the code so that it works properly. In Hour 5, “Get Connected—Building Hyperlinks for Navigation and Further Exploration,” you’ll create a second page and add hyperlinks to your pages to make them interactive.

## Q&A

- Q.** *When I imported the content from the Word document, there was all this extra code attached to the heading to make it larger and bold. How come simply adding the `<h1>` tags has the same effect?*
- A.** All browsers interpret predefined styles in HTML in much the same way. All these styles are available from the Styles drop-down menu. If you don't specifically define these styles in your document, they display according to their default parameters. In later lessons, you will learn to change the default styles to your specification and to make new styles.
- Q.** *Why can't I simply cut and paste the content from a Word document straight into my web page?*
- A.** A word processor, such as Microsoft Word, inserts a large amount of hidden style code in documents. When you copy and paste text from these programs, all of that code quite literally tags along. One example of this is simple line breaks. When you press the Enter key in a word processor, you are actually inserting two line break codes rather than one. So, when you copy and paste the content to Expression Web 2, an extra line of empty space appears between each paragraph and you have to delete them manually. There are many ways of getting around this problem. One is to copy the content into a pure text editor such as Notepad first and copy it from there into Expression Web 2. But this approach can cause a lot of confusion and extra work because it strips away all the style information and gives you only plain text. There is nothing technically wrong with cutting and pasting content from word processors to Expression Web 2, but it creates a lot of extra work.

## Workshop

The Workshop has quiz questions and exercises to help you put to use what you just learned. If you get stuck, the answers to the quiz questions are in the next section. But try to answer them first. Otherwise you'll only be cheating yourself.

## Quiz

1. *Why should you use the Import File feature when importing content from an external file?*
2. *In Code view, how do you tell the browser to treat one line of text differently from another?*

## Answers

1. Other applications such as word processors apply styling such as font type and size to each individual object. When you cut-and-paste this content into your web page all this styling content comes along with it causing the markup to become cluttered with unnecessary code. The Import File feature strips away most of this extra code and makes it easier for you to format the content properly for the web.
2. To separate and group content HTML uses *tags*. Any content placed between a beginning and an end tag is considered by the browser to be styled by this tag and its attributes and is displayed accordingly. When you apply a particular style to content in Design view you are really wrapping this content in specific tags.

## Exercise

Find a Word document you already have on your computer or write a new one. Make sure it has several headings and subheadings along with bold and italicized content and a few lists. With the `default.html` file open in Expression Web 2, choose Save As under File and select a new name like `myTest.html` and save it. Delete all the content in your new file and import the content from the Word document. Clean up the code so that it has only styles you can find in the Styles drop-down menu, and all bold and italicized content is strong and emphasized instead. Save the file and test it in your browser to verify that it works properly.

# Index

## Symbols

:active pseudoclass, 173  
:focus pseudoclass, 174  
:hover pseudoclass, 174  
:link pseudoclass, 174  
.psd files, importing images, 219-220  
:visited pseudoclass, 174

## A

absolute hyperlinks, 73  
absolute option (position attribute), 230  
Accessibility Properties dialog, 87  
accessing toolbars, 9  
actions, 268  
adding  
    columns to tables, 130  
    Common Toolbar items, 9-10  
    Insert File command to Common Toolbar, 58  
    rows to tables, 130  
adjusting vertical alignment in tables, 129  
Adobe Photoshop files, importing, 216-220  
advanced box button, creating, 259-261  
Advanced Button form control, 359  
advanced publishing settings, Remote Web Site Properties dialog

Optimize HTML tab, 448-449  
Publishing tab, 449-451  
animation, Silverlight, 382  
appearance  
    of cells, changing, 135-136  
    of tables, changing with Autoformat option, 137  
Appearance tab (Picture Properties dialog), 90-93  
Apple Safari, 48  
Apply Style task pane, 13-14, 155  
    color coding, 156  
    styles, creating, 144-146  
applying  
    background images to styles, 209-211  
    classes to tags, 189-190  
    CSS classes to content, 164-165  
    CSS resets, 226  
    external style sheets to web pages, 198-200  
    font family to whole documents, 149-152  
    framework to web pages, 233-239  
    images as background, 206-207  
    styles  
        to links, 174-176  
        to small selections, 148  
        to website, 29  
        with box model, 177-179

with CSS tools, 157-158  
ASP.NET, 386, 415  
    applications, 416  
    content pages, 423-426  
    content regions, 420-423  
    master page, 418-419  
        displaying in Code view, 421  
        modifying, 430-431  
    Portfolio Starter Kit, configuring contact page, 432-433  
attaching DWTs to existing web pages, 348-349  
attributes of Background category (body style), 207-208  
Auto Thumbnail button (Pictures toolbar), 98, 104-105  
Autoformat option, changing appearance of tables, 137

## B

Background category (body style), attributes, 207-208  
background images, 206-207  
    and Box model, 212  
    applying to styles, 209-211  
    sliding doors, creating text-free buttons, 262-264  
    stacking, 212-213  
basic box button, creating, 256, 259  
behaviors, 267  
    adding to website, 30-31  
    Call Script, 272

## behaviors

Change Property, 273-275  
 Change Property Restore, 276  
 Check Browser, 276-278  
 Check Plug-In, 278-279  
 comparing with CSS, 275-276  
 drop-down menus, creating, 323-324  
 events, onmouseover, 271-272  
 Go to URL, 279  
 Jump Menu, 280-281  
 Jump Menu Go, 281-282  
 modifying, 270  
 Open Browser Window, 282-284  
 Play Sound, 284  
 Popup Message, 285  
 Preload Images, 270, 285  
 Set Text, 286  
 Set Text of Frame, 301-303  
 Set Text of Layer, 310  
 Swap Image behavior, creating, 268, 270  
 variables, modifying, 270

**Block Selection visual aid, 45-47**

**blogs, WordPress, 339**

**body style, Background category attributes, 207-208**

**bookmarks**  
 creating, 76-78  
 temporary, 120

**box model, 176**  
 and background images, 212  
 content, styling, 177-179  
 dimensions, calculating, 177

**Box tab (CSS Sculptor), 242**

**Bring Forward/Backward button (Pictures toolbar), 98**

**broken hyperlinks, fixing, 36**

**Browsed Pages option (Insert Hyperlink dialog), 75**

**browsers**  
 cross-browser compatibility, enabling with embed, 381-382  
 CSS drop-down menu support, 333  
 web pages, testing, 47-48

**building framework with CSS Sculptor, 240-241**  
 Box tab, 242  
 Design tab, 244  
 Layout tab, 241  
 Output tab, 246  
 Print tab, 245  
 Type tab, 243

**bulleted lists, 115**  
 converting to number lists, 116  
 creating, 47

**bullets, images as, 214-216**

**buttons, 252**  
 advanced box button, creating, 259-261  
 creating with CSS, 256-259  
 Interactive Buttons  
 creating, 252, 255-267  
 editing, 255-256  
 saving, 255-256  
 text-free, creating with sliding doors, 262-264

## C

calculating dimensions for box model, 177

**Call Script behavior, 272**

**cascade, 136**

**cell padding, 134**

**Cell Properties dialog, changing appearance of cells, 135-136**

**cells**  
 appearance of, changing, 135-136  
 deleting, 131  
 merging/splitting, 131

**centering**  
 images with CSS classes, 165-166  
 pages with ID, 170-172

**Change Property behavior, 273-275**

**Change Property Restore behavior, 276**

**changing content outside editable regions, 353**

**Check Browser behavior, 276-278**

**Check Plug-In behavior, 278-279**

**circular hotspots, creating, 106**

**classes, 173**  
 applying  
 to content, 164-165  
 to tags, 189-190  
 embedding within classes, 172  
 images, centering, 165-166  
 styles, customizing, 168

**CMSs (Content Management Systems), 338-339**

**code comment, 120**

**Code view, 7, 10**

ASP.NET master page, displaying, 421

classes, applying to tags, 189-190

code errors, 56

content pages, displaying, 424-426

CSS code, displaying, 185

embed, 382

files, editing, 53-57

forms, 371-373

hyperlinks, 112-114

images, 114-115

imported text, cleaning up, 60-62

IntelliSense, 116-118

styles, creating with IntelliSense, 188-189

toolbar

Find Matching Tag button, 120

Follow Hyperlink button, 119

Insert Comment button, 120

Insert Tag button, 120

List Members button, 118

Next/Previous Bookmark button, 119

Next/Previous Code Hyperlink button, 119

Select Block button, 120

Toggle Bookmark button, 119

**collapsing menus, 7**

**Color button (Pictures toolbar), 99**

**color coding in Apply and Manage Styles task panes, 156****columns**

adding to tables, 130

deleting, 131

resizing, 132

**Common toolbar, 8**

Insert File command, adding, 58

items, removing/adding, 9-10

New icon, 42

**comparing CSS and behaviors, 275-276****contact form (Portfolio Starter Kit), configuring, 432-433****contact pages, creating with PHP, 400****content versus styles, 186****content pages, 423-426****content regions, 419-423****converting unordered lists to ordered lists, 116****creating**

buttons

- advanced box button, 259-261
- using CSS, 256-259

contact pages with PHP, 400

drop-down menus using CSS, 324-333

DWTs, 341-343

editable keywords with DWTs, 350-352

email forms with PHP, 403-408

external style sheets, 193-194

folders, 86

framework from boxed parts, 227-230

Home links, 72-73

hotspots, 106-107

HTML forms, 358-364

**hyperlinks**

- bookmarks, 76-78
- E-mail Address links, 78-79
- external links, 73-75
- internal links, 69-71

inline frames, 298-301

Interactive Buttons, 252, 255, 267

layers, 304-307

layers with inline frame, 307-309

layers-based drop down menu

- main menu, 316

- submenu, 317-321

nonstandard tables with Table Layout tool, 138-139

styles with Apply Styles task pane, 144-146

styles with IntelliSense, 188-189

Swap Image behavior, 268-270

tables, 127-129

- headings, 129

thumbnails, 104-105

web pages with DWTs, 343-345

websites, 42-44

- Block Selection visual aid, 46-47

- bulleted lists, 47

- from One-Page Website template, 43

**Crop button (Pictures toolbar)**

**Crop button (Pictures toolbar),**  
99, 102-103

**cropping images, 102-103**

**cross-browser compatibility,**  
**enabling with embed, 382.**  
*See also* CSS tools

**CSS (Cascading Style Sheets),**  
126, 150

advanced box button, creat-  
ing, 259-261

Background category (body  
style), attributes, 207-208

background images

applying to styles,  
209-211

stacking, 212-213

boxes, building, 30

buttons, creating, 256-259

classes

applying, 164-165

embedding within classes,  
172

images, centering,  
165-166

styles, customizing, 168

code, viewing, 184-185

comparing with behaviors,  
275-276

drop-down menus, creating,  
324-333

external style sheets

applying to web pages,  
198-200

creating, 193-194

moving styles to/from,  
195-198

files, storing, 37

IDs, 169-172

images, applying as  
background, 206-207

layouts, creating, 30

position attribute

absolute option, 230

fixed option, 231

inherit option, 233

relative option, 232

static option, 233

prepackaged layouts, 226

Quick Tag Tools, 153-154

resets, applying, 226

style sheets, 37

styles

applying to fonts, 146

overriding, 151-153

**CSS Properties task pane, 12,**  
**154-155**

**CSS Sculptor, framework, build-  
ing, 240**

Box tab, 242

Design tab, 244

Layout tab, 241

Output tab, 246

Print tab, 245

Type tab, 243

**CSS tools**

Apply Styles task pane, 155

Manage Styles task pane,  
155

styles, applying, 157-158

**customizing**

styles within classes, 168

workspace, 15-17

**D**

**Default.aspx page, 423**

displaying in Code view,  
424-426

**default.html, 44**

**deleting**

cells from tables, 131

columns from tables, 131

rows from tables, 131

**Design Portfolio Starter Kit,**  
**installing, 416-418**

**Design tab (CSS Sculptor), 244**

**Design view, 7, 11, 45**

Code view changes,  
displaying, 55

images, importing, 87-89

imported text, cleaning up,  
61-62

tables, creating, 127-129

**designing web pages, 223-225**

**destructive image editing tools,**  
**100**

**dialog boxes**

Accessibility Properties, 87

Cell Properties, changing  
appearance of cells,  
135-136

Insert Hyperlink, 70

Browsed Pages option, 75

Interactive Buttons, 253

Open Sites, 53

Picture Properties, 89

Appearance tab, 90-93

General tab, 89-90

Table Properties, changing  
appearance of tables,  
133-135

**dimensions, calculating for box  
model, 177**

**displaying ASP.NET master  
page in Code view, 421**

**divs, 168**

**domain name, obtaining, 438**

**downloading Design Portfolio  
Starter Kit, 418**

- Drop-Down Box form control, 359**
  - drop-down menus, 32**
    - basing on layers and behaviors, 323-324
    - managing, 322-323
  - DWTs, 338-339**
    - attaching to existing web pages, 348-349
    - changing content outside editable regions, 353
    - creating, 341-343
    - editable keywords, creating, 350-352
    - editing, 346-347
    - web pages, creating, 343-345
  - dynamic web pages, 395-396**
  - dynamic websites, 424**
- ## E
- E-mail Address links, creating, 78-79**
  - editable keywords, creating with DWTs, 350-352**
  - editing**
    - content outside editable regions, 353
    - DWTs, 346-347
    - files in Code view, 53-57
    - frames, 293-296
    - images, 89, 100
    - Interactive Buttons, 255-256
  - email forms, 32-33**
    - creating with PHP, 403-404
    - empty messages, filtering with PHP, 410-411
    - functionality, adding with PHP, 405-408
    - landing pages, creating with PHP, 409-410
  - embedding classes within classes, 172**
  - Empty Web Site template, 42**
  - enabling cross-browser compatibility using embed, 382**
  - errors, viewing in Code view, 56**
  - events, 267**
    - modifying, 270
    - onmouseover, 271-272
  - Expression Blend 2, 383**
  - Expression Development Server, 386**
    - PHP scripts, testing, 397-400
  - Expression web site, installing Design Portfolio Starter Kit, 416-418**
  - external links, 68**
    - creating, 73-75
    - linked pages, opening in new window, 75-76
  - external sources, modifying in Portfolio Starter Kit, 428-429**
  - external style sheets**
    - applying to web pages, 198-200
    - moving styles to/from, 195-198
  - external style sheets, creating, 193-194**
- ## F
- files, CSS, storing, 37**
  - filtering empty email messages with PHP, 410-411**
  - Find and Replace tool, 62**
  - Find Matching Tag button (Code view toolbar), 120**
  - Firefox, 48**
  - fixed option (position attribute), 251**
  - fixing broken hyperlinks, 36**
  - Flash, 33, 375-376**
    - embed, 381-382
    - photo galleries
      - customizing, 390-391
      - publishing, 377-379
  - Flash SWF Properties dialog, 380**
  - Folder List pane, 12, 101**
  - folders, creating, 86**
  - Follow Hyperlink button (Code view toolbar), 119**
  - font family, applying to documents, 149-152**
  - font sizes, 146**
  - form controls, 358**
  - forms**
    - email
      - adding functionality with PHP, 405-408
      - creating with PHP, 403-404
      - filtering with PHP, 410-411
      - landing pages, creating with PHP, 409-410
    - FPSE, 32-33
    - HTML, 357
      - creating, 358-364
      - in Code view, 371-373
      - sending results to email, 366-370
      - server-side scripts, 365
    - PHP, 32-33
      - results, uses, 370
    - FPSE (FrontPage Server Extensions) forms, 32-33
  - framed navigation, 297-298**
  - frames, 289, 297, 304**

**frames**

editing, 293-296

inline, 31

creating, 298-301

Silverlight applications,  
inserting, 388-389

Set Text of Frame behavior,  
301-303

web pages, creating, 291-293

**framework**

applying to website, 233-239

building with CSS Sculptor,  
240-241

Box tab, 242

Design tab, 244

Layout tab, 241

Output tab, 246

Print tab, 245

Type tab, 243

creating from boxed parts,  
227-230

**FrontPage Server Extensions,  
441-442**

**FTP, publishing your website,  
443-447**

**G**

**General tab (Picture Properties  
dialog), 89-90**

**GIF (Graphics Interchange  
Format) image format, 84**

**globally renaming styles,  
191-192**

**Go to URL behavior, 279**

**Group Box form control, 359**

**H**

**headings, adding to tables, 129**

**Home links, creating, 72-73**

**home page, 44**

**hotspot buttons (Pictures  
toolbar), 99, 106-107**

**hotspots, creating, 106-107**

**HTML**

and PHP, 396

contact pages, creating, 400

default.html, 44

forms, 357

creating, 358-364

results, sending to email,  
366-370

server-side scripts, 365

tags, 55-57

**hyperlinks, 67, 252**

absolute, 73

bookmarks

creating, 76-78

temporary, 120

E-mail Address links, creating,  
78-79

external links, creating, 73-75

fixing, 36

home links, creating, 72

in Code view, 112-114

inline links, 69

internal links, creating, 69-73

linked pages, opening in new  
window, 75-76

previewing, 28

relative, 73

root-relative, 73

ScreenTips, adding, 71

styles, applying, 174-176

**Hyperlinks report, 34**

**I**

**IDs, 169-172**

**IE Hacks, 49**

**image formats, 84-85**

**images**

adding to photo gallery, 391

applying as background,  
206-207

as list bullets, 214-216

backgrounds

applying to styles, 209,  
211

sliding doors, 262

stacking, 212-213

centering with CSS classes,  
165-166

cropping, 102-103

editing, destructive image  
editing tools, 100

hotspots, creating, 106-107

importing, 86-89

from Photoshop files,  
216-220

in Code view, 114-115

inserting into web page, 85

previewing, 29

replaced elements, 115

resampling, 103

resizing, 101

separating from text, 93

thumbnails, creating,  
104-105

**Import Site Wizard, 24-26**

**Import Web Site template, 42**

**imported text, cleaning up**

in Code view, 60-62

in Design view, 61-62

**importing**

- Adobe Photoshop files, 216-220
- images, 86-89
- new web page, 68
- style sheets in HTML files, 195
- styled text, 57-58
- websites, 24-26

**inherit option (position attribute), 233****inline frames, 31**

- creating, 298-301
- Silverlight applications, inserting, 388-389

**inline links, 69****inline styles, 147, 186-187****Input (Button) form control, 359****Input (Check Box) form control, 359****Input (File) form control, 359****Input (Hidden) form control, 359****Input (Image) form control, 359****Input (Password) form control, 359****Input (Radio) form control, 359****Input (Reset) form control, 360****Input (Submit) form control, 360****Input (Text) form control, 360****Insert Comment button (Code view toolbar), 120****Insert File command, adding to Common Toolbar, 58****Insert Hyperlink dialog, 70**  
Browsed Pages option, 75**Insert Picture button (Picture toolbar), 100****Insert Picture from File button (Pictures toolbar), 98****Insert Tag button (Code view toolbar), 120****inserting**

- Flash movies, 378
- images in web pages, 85
- Silverlight applications in web page, 383-388

**installing**

- Design Portfolio Starter Kit, 416, 418
- Expression Web 2, 5
- PHP, 397-400

**IntelliSense, 116-118**

- styles, creating, 188-189

**Interactive Buttons**

- creating, 252, 255, 267
- editing, 255-256
- saving, 255-256

**Interactive Buttons dialog, 253****internal links, 68**

- creating, 69-73
- ScreenTips, adding, 71

**Internet Explorer, 48****J-K****JavaScript, 267****JPEG (Joint Photographic Experts Group) image format, 84****Jump Menu behavior, 280-281****Jump Menu Go behavior, 281-282****L****Label form control, 360****landing pages, adding to email forms with PHP, 409-410****layers, 31**

- creating, 304-307
- drop-down menus, creating, 323-324

**Set Text of Layer behavior, 310**

- with inline frame, creating, 307-309

**layers-based drop down menus**

- main menu, creating, 316
- submenu menu, creating, 317-321

**layout boxes, styling, 228-230****Layout section (Picture Properties dialog, Appearance tab), 91****Layout tab (CSS Sculptor), 241****layouts, creating with CSS, 30****Less Brightness button (Pictures toolbar), 99****linking style sheets to HTML files, 195****links, applying styles with pseudoclasses, 174-176****List Members button (Code view toolbar), 118****local computer, website publishing, 444****local websites, 26-27****M****Manage Styles task pane, 13-14, 155**

- color coding, 156

- CSS code, viewing, 184-185

**managing, drop-down menus, 322-323****markup languages, 44****master page (ASP.NET), modifying, 430-431****menu bar, 6****menus**

- collapsing, 7

## menus

- drop-down, 32
  - basing on layers and behaviors, 323-324
  - creating with CSS, 324-333
  - managing, 322-323
- layers-based drop down menu, creating
  - main menu, 316
  - submenu menu, 317-321
- merging cells, 131
- Meyer, Eric A., 150, 226, 240
- modifying
  - ASPNET master page, 430-431
  - behaviors, 270
- More Brightness button (Pictures toolbar), 99
- Mozilla Firefox, 48
- MSDN Library page, 290

## N

- New icon (Common toolbar), 42
- Next/Previous Bookmark button (Code view toolbar), 119
- Next/Previous Code Hyperlink button (Code view toolbar), 119
- numbered lists, 115

## O

- obtaining domain name, 438
- One Page Web Site template, 42-43
- onmouseover event, 271-272
- Open Browser Window behavior, 282-284
- Open Sites dialog, 53
- Open Source software, 339
- opening linked pages in new window, 75-76

- Opera browser, 48
- Optimize HTML tab (Remote Web Site Properties tab), 448-449
- ordered lists, 115
- orientation buttons (Pictures toolbar), 99
- Output tab (CSS Sculptor), 246
- overlapping background images, 212-213
- overriding styles with CSS, 151-153

## P

- photo galleries
  - customizing, 390-391
  - publishing, 377-379
- Photoshop files, importing, 216-220
- PHP, 395
  - and HTML, 396
  - contact pages, creating, 400
  - email forms
    - creating, 403-404
    - empty messages, filtering, 410-411
    - functionality, adding, 405-408
    - landing pages, creating, 409-410
  - forms, 32-33
  - installing, 397-400
  - scripts, testing, 397-400
  - variables, 402
- Picture Properties dialog, 89
  - Appearance tab, 90-93
  - General tab, 89-90
  - images, resizing, 101
- Pictures toolbar
  - Auto Thumbnail button, 98, 104-105
  - Bring Forward/Backward button, 98
  - Color button, 99
  - Crop button, 99, 102-103
  - hotspot buttons, 99, 106-107
  - Insert Picture button, 100
  - Insert Picture from File button, 98
  - Less Brightness buttons, 99
  - More Brightness buttons, 99
  - orientation buttons, 99
  - Resample Image button, 99
  - Restore button, 99
  - Set Transparent Color button, 99
- pixels, 146
- Play Sound behavior, 284
- PNG (Portable Network Graphics) image format, 85
- polygonal hotspots, creating, 106
- Popup Message behavior, 285
- Portfolio Starter Kit
  - contact form, configuring, 432-433
  - external sources, modifying, 428-429
- position attribute
  - absolute option, 230
  - fixed option, 251
  - inherit option, 233
  - relative option, 232
  - static option, 233
- positioning background on web page, 208
- Preload Images behavior, 270, 285
- prepackaged CSS layouts, 226
- previewing
  - Silverlight applications, 386

- websites, 27
  - hyperlinks, 28
  - images, 29
  - tables, 29
- Previous Code Hyperlink button (Code view toolbar), 119**
- Print tab (CSS Sculptor), 245**
- Program bar, 6**
- pseudoclasses, 173**
  - links, styling, 174-176
- Publishing tab (Remote Web Site Properties tab), 449-451**
- publishing your website**
  - FrontPage Server Extensions, 441-442
  - FTP, 443-444
  - on local computer, 444
  - WebDAV, 442
  - with FTP example, 445-447

## Q-R

- Quick Tag Tools, 153-154**
- recursive acronyms, PHP, 397
- refreshing Folder List task pane, 101
- relative hyperlinks, 73
- relative option (position attribute), 232
- Remote Web Site Properties dialog, 439**
  - Optimize HTML tab, 448-449
  - Publishing tab, 449, 451
- remote websites, 26-27**
- removing**
  - Common Toolbar items, 9-10
  - task panes from view, 15-17
- renaming styles globally, 191-192**
- repeating images as background, 207**

- replaced elements, 115**
- reports**
  - Hyperlinks, 34
  - Slow Pages, 34
- Resample Image button (Pictures toolbar), 99**
- resampling images, 103**
- resizing**
  - images, 101
  - rows and columns, 132
- Restore button (Pictures toolbar), 99**
- root-relative hyperlinks, 73**
- Rotate buttons (Picture toolbar), 99**
- rows**
  - adding to tables, 130
  - deleting, 131
  - resizing, 132

## S

- Safari, 48**
- saving Interactive Buttons, 255-256**
- ScreenTips, 9, 71**
- Select Block button (Code view toolbar), 120**
- selectors, pseudoclasses, 173-176**
- sending form results to email, 366-370**
- separating**
  - images from text, 93
  - content with boxes, 167-168
- server-side scripts, 365**
- Set Text behavior, 286**
- Set Text of Frame behavior, 301-303**
- Set Text of Layer behavior, 310**
- Set Transparent Color button (Pictures toolbar), 99**
- shortcuts, 9**
- Silverlight, 375, 382**
  - applications, 33
    - inserting in web page, 383-388
    - inserting with inline frames, 388-389
    - previewing, 386
  - photo galleries, customizing, 390-391
- Size section (Picture Properties dialog, Appearance tab), 92**
- sketching website layout, 224-225**
- SlideShowPro plugin, 377**
- sliding doors, creating text-free buttons, 262-264**
- Slow Pages report, 34**
- Split view, 12**
- splitting cells, 131**
- stacking background images, 212-213**
- standards-based sites, 4**
- static option (position attribute), 233**
- static pages, 395**
- static websites, 424**
- Status bar, 15**
- storing CSS files, 37**
- style IDs, 169-172**
- style sheets, 11, 37, 195**
- styled text, importing, 57-58**
- styles, 147**
  - applying
    - to small selections, 148

## styles

- to website, 29
- with box model, 177-179
- with CSS tools, 157-158
- creating, 144, 146
  - with IntelliSense, 188-189
- customizing within classes, 168
- font sizes, 146
- inline styles, 186-187
- moving to/from external style sheets, 195-198
- overriding with CSS, 151-153
- renaming globally, 191-192
- versus content, 186

**Swap Image behavior, creating, 268, 270**

## T

- Table Layout tool, creating nonstandard tables, 138-139**
- Table Properties dialog, changing appearance of tables, 133-135**
- tables**
  - appearance of, changing, 133-137
  - cells
    - appearance of, changing, 135-136
    - deleting, 131
    - merging/splitting, 131
  - columns
    - adding, 130
    - deleting, 131
    - resizing, 132
  - creating, 127-129
  - headings, creating, 129
  - nonstandard, creating with Table Layout tool, 138-139

- previewing, 29
- rows
  - adding, 130
  - deleting, 131
  - resizing, 132
- verical alignment, adjusting, 129

### Tag Properties subpane, 12

#### tags

- box model, 176
  - content, styling, 177-179
  - dimensions, calculating, 177
- classes, applying, 189-190
- renaming globally, 191-192

#### tags (HTML), 55-57

#### task panes

- Apply Styles pane, 13-14
- CSS Properties subpane, 12
- Folder List pane, 12
- Manage Styles pane, 13-14
- removing from view, 15-17
- Tag Properties subpane, 12
- Toolbox pane, 13

#### Task Panes menu, adding/removing task panes from view, 15-17

#### templates

- DWTs, 338-339
  - attaching to existing web pages, 348-349
  - changing content outside editable regions, 353
  - creating, 341-345
  - editing, 346-347
- Empty Web Site option, 42
- Import Web Site option, 42
- One Page Web Site, 42-43

#### temporary bookmarks, 120

#### testing

- PHP scripts, 397-400
- web pages, 47-48

#### text, separating images from, 93

#### Text Area form control, 360

#### text-free buttons, creating with sliding doors, 262-264

#### thumbnails, creating, 104-105

#### Toggle Bookmark button (Code view toolbar), 119

#### toolbars

- accessing, 9
- Code view
  - Find Matching Tag button, 120
  - Follow Hyperlink button, 119
  - Insert Comment button, 120
  - Insert Tag button, 120
  - List Members button, 118
  - Next/Previous Bookmark button, 119
  - Next/Previous Code Hyperlink button, 119
  - Select Block button, 120
  - Toggle Bookmark button, 119
- Pictures toolbar
  - Auto Thumbnail button, 98, 104-105
  - Bring Forward/Backward button, 98
  - Color button, 99
  - Crop button, 99, 102-103
  - hotspot buttons, 99, 106-107
  - Insert Picture button, 100

- Insert Picture from File button, 98
- Less Brightness button, 99
- More Brightness button, 99
- orientation buttons, 99
- Resample Image button, 99
- Restore button, 99
- Set Transparent Color button, 99

#### Toolbox task pane, 13

- Form Controls section, 358

#### Type tab (CSS Sculptor), 243

## U-V

#### unordered lists, 115

- converting to ordered lists, 116
- drop-down menus, creating, 324-333
- images, using as bullets, 214-216

#### URLs (uniform resource locators), 70

- external links, creating, 73-75

#### variables

- modifying, 270
- PHP, 402

#### vertical alignment, adjusting in tables, 129

#### View pane, 7

#### viewing

- CSS code, 184-185
- website in Expression Web 2, 34-35

#### views, 7

- Code view, 10
  - classes, applying to tags, 189-190
  - CSS code, viewing, 185
  - files, editing, 53-57
  - forms, 371-373
  - hyperlinks, 112-114
  - images, 114-115
  - imported text, cleaning up, 60-62
  - Intellisense, 116-118
  - toolbar, 118
- Design view, 11, 45
  - displaying changes in Code view, 55
  - images, importing, 87-89
  - imported text, cleaning up, 61-62
  - tables, creating, 127-129

#### Split view, 12

#### visual aids, Block Selection, 45-47

#### Visual Aids menu, 45

## W

#### W3C (World Wide Web Consortium), 4

#### web browsers, 47

- CSS drop-down menu support, 333

#### web pages

- creating with DWTs, 343-345
- designing, 223-225
- editing with DWTs, 346-347
- importing, 68
- testing, 47-48

#### web-hosting services, 438-439

#### WebDAV, publishing your website, 442

#### websites

- behaviors, adding, 30
- creating, 42-44
  - Block Selection visual aid, 46-47
  - from One-Page Website template, 43
- designing, 223-225
- framework, creating, 227-230, 233-239
- home page, 44
- hyperlinks, previewing, 28
- images, previewing, 29
- importing, 24-26
- local, 26-27
- previewing, 27
- remote, 26-27
- styles, applying, 29
- tables, previewing, 29
- viewing in Expression Web 2, 34-35

#### WordPress, blogs, 339

#### workspace, customizing, 15-17

#### Wrapping Style options (Picture Properties dialog, Appearance tab), 91

#### WYSIWYG (What You See Is What You Get), 4

## X-Y-Z

#### XAML files, 376

#### YouTube, 382