Microsoft

Step by Step

Microsoft® Expression® Web 4

Build *exactly* the skills you need.
Learn at the pace *you* want.

Get **practice files** and **online edition** of this book
See back

Chris Leeds

The smart way to learn Microsoft® Expression® Web 4—one step at a time!

Experience learning made easy—and quickly teach yourself how to create dynamic, standards-based Web sites. With *Step by Step*, you set the pace—building and practicing the skills you need, just when you need them!

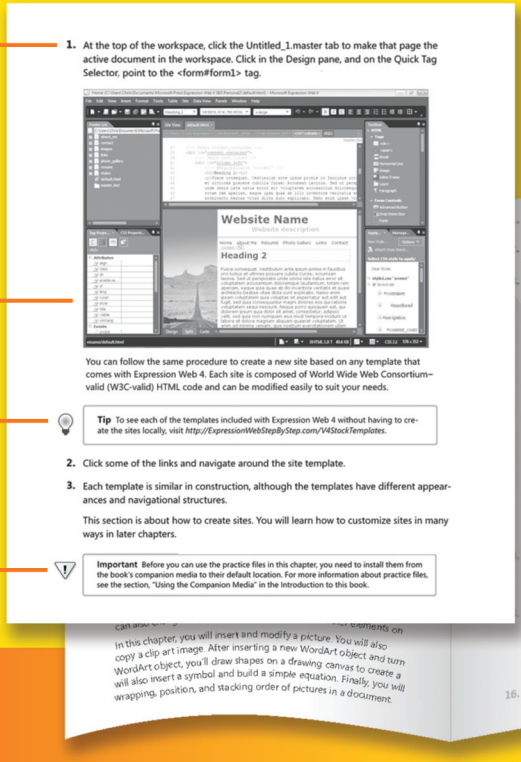
- Build your site with customizable templates—or create your own
- Adjust settings in Expression Web 4 to create different sites
- Add Photoshop files, Microsoft Silverlight® video, and .wmv files
- Use CSS, HTML, and XHTML to manage site style and accessibility
- Use server-side scripting such as ASP.NET and PHP
- Choose the right publishing method for your scenario

1 **Follow the easy numbered steps**

2 **Screen images help you check your work**

3 **Tips offer helpful hints**

4 **Build your skills hands-on using ready-made practice files**



Your *Step by Step* digital content includes:

- All the book's **practice files**—ready to download and put to work. *See "Code Samples," inside.*
- Fully searchable **online edition** of this book—with unlimited access on the Web. *Free online account required; see inside back.*
- **Video tutorials** illustrating important procedures and concepts. *Details inside.*

ISBN: 978-0-7356-xxxx-x

U.S.A. $44.99
Canada $51.99
[*Recommended*]

Web Design/Microsoft Expression Web

Is this the right book for you?
For help, see inside the back cover.

Microsoft

Microsoft®

Microsoft® Expression® Web 4 Step by Step

Chris Leeds

Copyright © 2010 by MediaCarbon, Inc.

All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher.

Unless otherwise noted, the example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted herein are fictitious. No association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred.

Microsoft Press titles may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://my.safaribooksonline.com*). For more information, contact our corporate/institutional sales department: (800) 998-9938 or *Microsoft Press, Inc.*

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without express written permission of Microsoft Press, Inc.

ISBN: 978-0-7356-3902-7

2 3 4 5 6 7 8 9 10 LSI 7 6 5 4 3 2

Microsoft, Microsoft Press, the Microsoft Press brand, Access, ASP.NET, DirectX, DreamSpark, Deep Zoom, Excel, Expression Blend, Expression Design, Expression Encoder, Expression Studio, Expression Web, FrontPage, .NET, Office, Silverlight, SQL Server, Visual Basic, Visual C++, Visual C#, Visual Studio, Visual Web Developer, Web Platform, WebsiteSpark, and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

This book expresses the author's views and opinions. The information contained in this book is provided without any express, statutory, or implied warranties. Neither the author, Microsoft Corporation, nor their respective resellers, or distributors will be held liable for any damages caused or alleged to be caused either directly or indirectly by this book.

Acquisitions and Developmental Editor: Russell Jones
Production Editor: Teresa Elsey
Editorial Production: Online Training Solutions, Inc.
Technical Reviewers: Kathleen Anderson and Gerry Tiegrob
Illustrator: Robert Romano
Indexer: Ellen Troutman Zaig
Cover: Karen Montgomery

[2012-11-30]

Contents at a Glance

Introduction xi
1 Understanding How Expression Web 4 Works 1
2 Capitalizing on Expression Web 4 Functionality 47
3 Capitalizing on the Template Options in Expression Web 4 107
4 It's All About Content 157
5 Understanding Validity and Accessibility 231
6 Creating a Web Site from Scratch 257
7 Adding Client-Side Functionality 323
8 Adding Functionality with jQuery and PHP 351
9 Adding Functionality with ASP.NET and AJAX 381
10 Managing and Publishing Your Work 435

Table of Contents

Acknowledgments ix
Introduction xi

1 **Understanding How Expression Web 4 Works** 1
Understanding the Expression Web 4 Interface 2
Opening a Site 9
Using Site Views 14
Opening a Page 20
Using Page Views 22
Using the Snapshot Panel 29
Using Visual Aids 33
Using Browser Preview 36
Using SuperPreview 41
Key Points 46

2 **Capitalizing on Expression Web 4 Functionality** 47
Changing Site Settings 48
Changing Application Options 54
Changing Page Editor Options 61
Using the Expression Development Server 65
Using PHP with the Expression Development Server 71
Creating a New Site with Expression Web 80
Creating a New Site from an Expression Web Template 80
Creating an Empty Site and Importing Files and Folders 91
Using the Import Site Wizard 96
Configuring Add-ins 101
Key Points 105

3 **Capitalizing on the Template Options in Expression Web 4** 107
Understanding Template Concepts 108
Using Dynamic Web Templates 109
Using the Include Page Feature 128
Using ASP.NET Master Pages 136
Using ASP.NET Web User Controls 145
Using PHP Include Files 150
Key Points 156

4 **It's All About Content 157**
Using Tables Properly 158
Using Lists to Group Information 163
Using Semantic Markup 173
Styling the Presentation of Your Content 177
Adding Images to a Web Page 188
Editing Images with Expression Design 198
Using Photoshop Files in a Web Page 209
Using Silverlight Video in a Web Page 213
Using Deep Zoom Composer Projects in a Web Page 222
Key Points 229

5 **Understanding Validity and Accessibility 231**
Understanding and Changing a DOCTYPE 232
Verifying and Addressing W3C Validity 238
Verifying and Addressing Accessibility 244
Styling Text with Cascading Style Sheets 247
Key Points 256

6 **Creating a Web Site from Scratch 257**
Working with a Graphics Template 257
Creating an HTML Layout 272
Styling Major HTML Elements 284
Designing the Site Architecture and Navigation 297
Styling for Alternative Media 316
Key Points 322

7 **Adding Client-Side Functionality 323**
Understanding Server-Side vs. Client-Side Scripting 323
Using Layers and Behaviors 324
Using Data View in an HTML Page 332
Creating and Using HTML Forms 343
Key Points 349

8 **Adding Functionality with jQuery and PHP 351**
Using jQuery in Expression Web 351
Using the Expression Web PHP Tools 362
Key Points 379

9 **Adding Functionality with ASP.NET and AJAX 381**
Using ASP.NET Tools in Expression Web 382
The ASP.NET Segment of the Toolbox 382
Converting a DWT to a Master Page 383
Using Site Navigation Controls 400
Using the AdRotator Control 411
Linking to Data Sources and Using Data Controls 415
Using ASP.NET Ajax Features 430
Key Points 434

10 **Managing and Publishing Your Work. 435**
Considering Hosting Requirements 436
Reseller Accounts 437
Finding and Registering a Domain Name 437
Using Microsoft Office Live Small Business 439
Understanding the Publishing Protocols Available in Expression Web 439
Considering Security Requirements 441
Publishing with FTP 441
Publishing Using HTTP (FrontPage Server Extensions) 449
File System Publishing 454
Managing a Web Site 459
Backing Up Server-Based Sites 459
Editing Server-Based Sites 465
Using Site Settings 472
Publishing Selectively 478
Using Subsites 485
Import and Export Web Packages 489
Use SEO Reports 493
Key Points 497

Index 499

About the Author 521

What do you think of this book? We want to hear from you!

Microsoft is interested in hearing your feedback so we can continually improve our books and learning resources for you. To participate in a brief online survey, please visit:

www.microsoft.com/learning/booksurvey/

Acknowledgments

I want to thank the readers of my previous edition of this book, and the visitors to my support site (www.expressionwebstepbystep.com). I'd like to include a special thanks to Russell Jones (O'Reilly Media, project editor), Kathleen Anderson and Gerry Tiegrob (technical reviewers), and all the people at Microsoft Press and O'Reilly Media who really had more to do with the physical manifestation of this book than I did. Last but not least, I'd like to thank the people on the Expression Web product team for their commitment to deliver a truly technology-agnostic Web design tool that makes modern, standards-based Web authoring so much easier than it used to be. Thank you all!

—Chris Leeds

Introduction

Microsoft Expression Web 4 is the newest Web editing and management application from Microsoft. It is a professional design tool used to create modern, standards-based sites that deliver superior quality on the Web.

From the perspective of a designer or developer, the modern Web bears only a slight resemblance to the Web of even as little as five years ago. Expression Web 4 is an effort by Microsoft to provide a tool that helps designers attain modern Web design standards and practices.

Expression Web 4 includes features that help ensure your output adheres to World Wide Web Consortium (W3C) validity standards, and that help you to cleanly separate content from presentation by taking advantage of the functionality and capabilities of cascading style sheets (CSS).

As you drill deeper into Expression Web 4, you will find tools to make working with Microsoft ASP.NET much more comfortable than ever before, such as the ASP.NET Development Server that installs with Expression Web and the ASP.NET Controls group in the Toolbox panel.

In addition, Expression Web helps you to easily use some of the features that ASP.NET offers, specifically, ASP.NET Master Pages, Navigation Controls, and Data Controls. It's refreshing that tools with this type of power are available in a user-friendly designer-centric application like Expression Web 4 as opposed to being limited to programmer-oriented tools such as Microsoft Visual Web Developer or Microsoft Visual Studio.

System Requirements

To perform the exercises in this book, your computer should meet the following requirements:

- Windows XP with Service Pack 3, Windows Vista, Windows 7, or Windows Server 2008 operating system
- PC with 1 GHz or faster processor
- 1 GB of RAM or more
- 2 GB or more of available hard disk space
- Microsoft .NET Framework 4
- Microsoft Silverlight 4
- A monitor capable of 1024×768 or higher resolution with 24-bit color

- Support for Microsoft DirectX 9 graphics with Windows Vista Display Driver Model (WDDM), 128 MB of graphics RAM or more, Pixel Shader 3.0 in hardware, 32 bits per pixel
- Internet access (additional fees may apply)
- Some product features require Firefox 3.0 or later and Internet Explorer 8

Actual requirements and product functionality may vary based on your system configuration and operating system.

Information for Readers Running Windows XP or Vista

The graphics and the operating system-related instructions in this book reflect the Windows 7 user interface, but you can also use a computer running Windows Vista or Windows XP with Service Pack 3 (SP3) installed.

Most of the differences you will encounter when working through the exercises in this book on a computer running Windows Vista or Windows XP relate to appearance rather than functionality. For example, the Windows 7 and Windows Vista Start buttons are round rather than rectangular and are not labeled with the word Start; window frames and window-management buttons look different; and if your system supports Windows Aero, the window frames might be transparent.

For the most part, differences in navigating to or through menus and dialog boxes in Windows Vista or Windows XP are small enough that you will have no difficulty in completing the exercises.

Who Is This Book For?

This book is intended for existing or new Web designers. Although the book intends to provide lessons to new Web designers, experienced practitioners will be able to quickly gain familiarity with the Expression Web 4 user interface and features. Former Microsoft FrontPage users should also find this book helpful in making a transition from FrontPage to Expression Web.

Assumptions

This book expects that you have at least a minimal understanding of basic computer use and file operations with your chosen operating system. No experience in Web design or development is required.

What's New in Expression Web 4

- **SuperPreview Online Service** The Microsoft Expression Web SuperPreview online service is a beta service that extends the capability of SuperPreview to include support for additional browsers and operating systems. For example, you can preview your pages in Apple Safari running on an OSX system "in the cloud."
- **SEO Checker** The SEO Checker feature of Expression Web analyzes your site against the best practices for getting the highest possible search engine rankings for your site. You can choose SEO options, display an SEO report, filter the results in the SEO report, and step forward and back through the list of results in the SEO report to see more detail for individual list items.

Acquiring Expression Web 4

You can purchase Expression Web 4 as part of Microsoft Expression Studio 4 Web Professional, which is an integrated group of applications that includes:

- **Expression Web 4 and SuperPreview** Expression Web is a professional design tool used to create engaging, Web-connected multimedia experiences for Windows. SuperPreview speeds browser compatibility testing by not only showing a high-fidelity rendering of how pages will look on different browsers, but it also identifies the element's tag, size, and position; applied styles; and location in the Document Object Model (DOM) tree so you can quickly fix the error.
- **Expression Design 4** Microsoft Expression Design 4 is the perfect companion to Microsoft Expression Blend or Expression Web. Use existing artwork or intuitive vector drawing tools to quickly build sophisticated vector and image assets. Seamlessly transfer them to your Expression Blend or Expression Web projects knowing that their fidelity and live effects will be maintained throughout the entire designer-developer workflow.
- **Expression Encoder 4** Microsoft Expression Encoder simplifies publishing video to Silverlight. Encode a wide array of video file formats, stream live from Web cams and camcorders, and screen capture on your computer. Make simple edits to video files and enhance your media with overlays and advertising. Choose encoding settings, select from a range of player templates, and publish rich media experiences with Silverlight.

You can also purchase Expression Web 4 as a component of Microsoft Expression Studio 4 Ultimate, which includes all that the professional version contains, plus these additional products:

- **Expression Blend 4** Utilizing the industry-proven technologies in Silverlight, WPF, XAML, Microsoft Visual C# and Microsoft Visual Basic, Expression Blend enables you to deliver applications that are stable, scalable, accessible, reliable, and highly secure, while maintaining optimum performance. SketchFlow, a feature of Expression Studio Ultimate, revolutionizes how quickly and efficiently you can demonstrate your vision for an application. It provides an informal and quick way to explore, iterate, and prototype user interface scenarios, allowing you to evolve your concepts from a series of rough ideas into a living and breathing prototype that can be made as real as a particular client or project demands.
- **Expression Encoder 4 Professional** The professional version of Expression Encoder contains more import and export codecs and unlimited screen recording duration, whereas the basic version of Expression Encoder is limited to 10-minute screen recordings.

Additionally, you can purchase both versions of Expression Studio as an upgrade—and the upgrade isn't limited to users of just Expression Web 3, but is available to owners of any previous Microsoft Expression product as well as any version of Adobe Creative Suite or Microsoft Visual Studio 2005 or later.

Beyond that, you can get Expression Studio as part of the Microsoft Partner program, at *https://partner.microsoft.com/40043420*, and through the Microsoft WebsiteSpark program at *http://websitespark.com*.

Finally, if you're a student, you can get Expression Studio 4 and a huge collection of other Microsoft development software through Microsoft DreamSpark, at *http://www.dreamspark.com*. With DreamSpark, students can download Microsoft developer and design tools at no charge, making it easier for them to learn the skills they need to excel both during school and after graduation.

Code Samples

Most of the chapters in this book include exercises that let you interactively try out new material learned in the main text. All sample projects are available for download from the book's page online:

http://aka.ms/639027/files

Installing the Code Samples

Unzip the file that you downloaded from the book's Web site to a location on your hard drive. In order for your screen to match the images in this book, we recommend that you unzip the folder to \User\Documents\Microsoft Press\Expression Web 4 SBS\.

Important Due to Operating System version, your personal preference, or other factors these files may be unzipped to a different location than specified above. Please note that the file and folder paths shown in screen shots are illustrative and shouldn't be taken literally.

What's in the Sample Code?

This book's sample code contains the main site called "SampleSite" with a page for each chapter of this book, several Deep Zoom Composer projects, videos, an Expression Design graphics template, and all the other assets required to complete the chapter exercises you will encounter.

Your Companion eBook

The eBook edition of this book allows you to:

- Search the full text
- Print
- Copy and paste

To download your eBook, please see the instruction page at the back of this book.

Organization of This Book

- Chapter 1, Understanding How Expression Web 4 Works
  - Identify and use the Expression Web 4 user interface elements
  - Open a site
  - Use Site Views
  - Open a page
  - Use page views
  - Use the Snapshot Panel
  - Use Visual Aids
  - Use browser preview
  - Use SuperPreview

- Chapter 2, Capitalizing on Expression Web 4 Functionality
  - Change site settings
  - Change Application Options
  - Change Page Editor Options
  - Use the Expression Development Server
  - Use PHP with the Expression Development Server
  - Create a new site with Expression Web
  - Use the Import Site Wizard
  - Configure add-ins
- Chapter 3, Capitalizing on the Template Options in Expression Web 4
  - Understand template concepts
  - Use Dynamic Web Templates
  - Use the Include Page feature
  - Use ASP.NET master pages
  - Use ASP.NET Web user controls
  - Use PHP include files
- Chapter 4, It's All About Content
  - Use tables properly
  - Use lists to group information
  - Use semantic markup
  - Style the presentation of your content
  - Add images to a Web page
  - Edit images with Expression Design
  - Use Photoshop files in a Web page
  - Use Silverlight Video in a Web page
  - Use Deep Zoom Composer projects in a Web page
- Chapter 5, Understanding Validity and Accessibility
  - Understand and change a DOCTYPE
  - Verify and address W3C validity
  - Verify and address accessibility
  - Address problems within HTML/XHTML code
  - Style text with cascading style sheets

- Chapter 6, Creating a Web Site from Scratch
  - Work with a graphics template
  - Create an HTML layout
  - Style major HTML elements
  - Design the site architecture and navigation
  - Style for alternative media
- Chapter 7, Adding Client-Side Functionality
  - Understand server-side versus client-side scripting
  - Use layers and behaviors
  - Use Data View in an HTML page
  - Create and use HTML forms
- Chapter 8, Adding Functionality with jQuery and PHP
  - Use jQuery in Expression Web
  - Use the Expression Web PHP tools
- Chapter 9, Adding Functionality with ASP.NET and AJAX
  - Use ASP.NET tools in Expression Web
  - Convert a DWT to a master page
  - Use site navigation controls
  - Use the AdRotator control
  - Link to data sources and use data controls
  - Use ASP.NET Ajax features
- Chapter 10, Managing and Publishing Your Work
  - Understand hosting requirements and publishing protocols
  - Find and register a domain name
  - Use Microsoft Office Live Small Business
  - Understand the publishing protocols available in Expression Web
  - Understand and evaluate security requirements
  - Publish with FTP, HTTP, or File System
  - Create a subsite
  - Import and export Web packages
  - Use SEO reports

Conventions and Features in This Book

This book has been designed to lead you step by step through all the tasks you are most likely to want to perform in Microsoft Expression Web 4. If you start at the beginning and work your way through all the exercises, you will gain enough proficiency to be able to create complex Web sites and pages. However, each topic is self-contained. If you have worked with another HTML or Web page editor, such as FrontPage, and simply need a quick guide to performing a task in Expression Web, or if after you complete all the exercises, you need a fast way to find the information to perform a procedure, the following features of this book will help you locate specific information:

- **Detailed table of contents** Scan this listing of the topics and sidebars within each chapter to quickly find the information you want.
- **Detailed index** Look up specific tasks, features, and general concepts in the index, which has been carefully crafted with the reader in mind.
- **Companion content** Install the practice files needed for the step-by-step exercises through this book's Web site. See the "Code Samples" section of this Introduction for instructions on getting and installing the practice files.

In addition, this book presents information using conventions designed to make the information readable and easy to follow.

- Most chapters include step-by-step exercises that you can follow to get guided exposure and practice to completing tasks.
- Each exercise consists of a series of tasks, presented as numbered steps (1, 2, and so on) listing each action you must take to complete the exercise.
- Boxed elements with labels such as "Note," "Tip," "Important," and so on provide additional information, directions for starting or ending the step-by-step exercises, or alternative methods for completing a step successfully.
- Text that you are supposed to type (apart from code blocks) appears in bold.
- A plus sign (+) between two key names means that you must press those keys at the same time. For example, "Press Alt+Tab" means that you hold down the Alt key while pressing Tab.

Errata and Book Support

We've made every effort to ensure the accuracy of this book and its companion content. If you do find an error, please report it on our Microsoft Press site:

1. Go to *www.microsoftpressstore.com*.
2. In the Search box, enter the book's ISBN or title.
3. Select your book from the search results.
4. On your book's catalog page, find the Errata & Updates tab
5. Click View/Submit Errata.

You'll find additional information and services for your book on its catalog page. If you need additional support, please email Microsoft Press Book Support at *mspinput@microsoft.com*.

Please note that product support for Microsoft software is not offered through the addresses above.

The author also maintains a support site for this book and Expression Web in general. Visit *www.expressionwebstepbystep.com*. Visit the site for links to this book's Facebook Fan Page, Live.com group, additional content, and contact information for the author. As a long-term member of the Microsoft MVP Community, Chris has been personally helping users with Microsoft Web design and development software for more than 10 years.

We Want to Hear from You

At Microsoft Press, your satisfaction is our top priority, and your feedback is our most valuable asset. Please tell us what you think of this book at:

http://www.microsoft.com/learning/booksurvey

The survey is short, and we read every one of your comments and ideas. Thanks in advance for your input!

Stay in Touch

Let's keep the conversation going! We're on Twitter: *http://twitter.com/MicrosoftPress*

Chapter 6

Creating a Web Site from Scratch

After completing this chapter, you will be able to:

- Work with a graphics template
- Create an HTML layout
- Style major HTML elements
- Design the site architecture and navigation
- Style for alternative media

In the previous chapters, you have seen some of the basic skills necessary to work efficiently in Microsoft Expression Web, but so far, you've worked only with existing graphics and templates.

In this chapter, you will build on the skills you already know by discovering ways to create graphics for your own custom design, and how to use those graphics to create a Web site template from scratch. These skills will be important whether you decide to create your own graphics or work with an existing Web site template that you customize.

Important Before you can use the practice files in this chapter, you need to download and install them from the book's companion content Web site to their default location. For more information about downloading and installing the practice files, see the "Code Samples" section at the beginning of this book.

Troubleshooting Graphics and operating system-related instructions in this book reflect the Windows 7 user interface. If your computer is running Windows XP or Windows Vista, and you experience trouble following the instructions as written, please refer to the "Information for Readers Running Windows XP or Vista" section at the beginning of this book.

Working with a Graphics Template

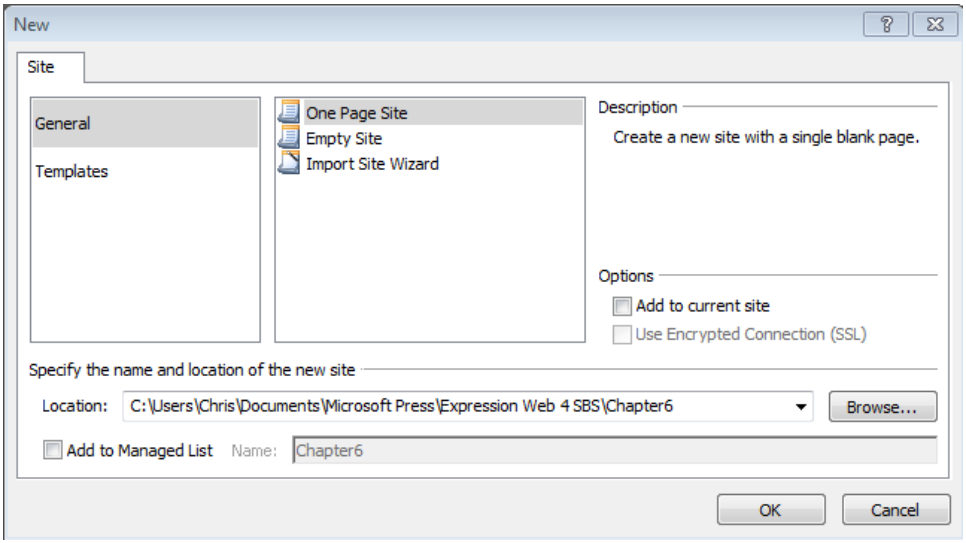
A designer can do only so much with colors and basic HTML/CSS layout techniques. At some point, graphics are required. Whether those graphics are used as backgrounds for various page areas, as buttons, icons, other elements, or a combination of these, designers must either create the graphics, or at the very least, be able to work with existing graphics to fine-tune them for use in their Web site layout.

Most Web site designs start with a sketch and an idea of what the end result will be. In this chapter, you will make a very simple Web site layout, and then create the graphics to go with it. The reason that you're going to work with the graphics first is simple: How can you create a layout if you don't know what it's going to look like?

Create and export a simple graphics template

Note Use the Chapter6.html page in the sample site. This site is located in the Documents\Microsoft Press\Expression Web 4 SBS\ folder. Open the SampleSite site by clicking Open on the Site menu, and then display the Chapter6.html page.

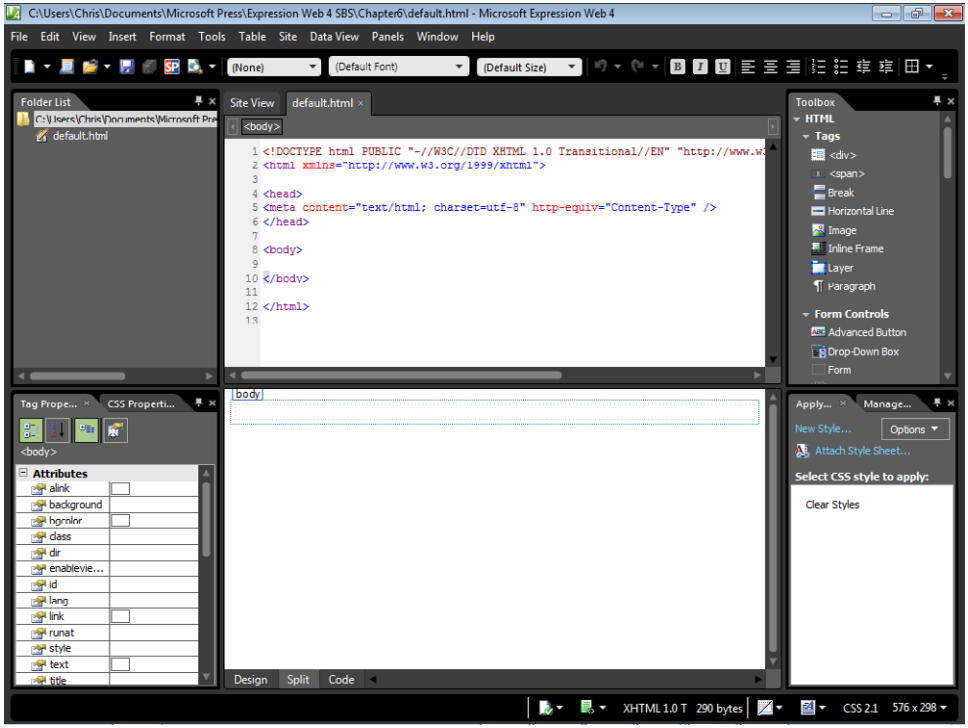
1. From the Site menu, click New. Select One Page Site, and in the location field, replace the automatically generated folder name by typing **Chapter6**. Save the site in the Documents\Microsoft Press\Expression Web 4 SBS\Chapter6 folder.



2. Click OK to begin the process of creating the new site.

   This book's Sample Site closes and Expression Web opens your newly created one-page site.

3. Double-click the default.html file to open it in the workspace. If your Editing window isn't already in Split view, click Split at the bottom of the workspace.



Notice that this is a truly minimal site. The Folder List panel contains just one file, and even the file open in the editing window has bare-minimum content. During the course of this chapter, all this will change radically.

4. From the Windows Start menu, click All Programs, select Microsoft Expression, and then click Microsoft Expression Design 4.

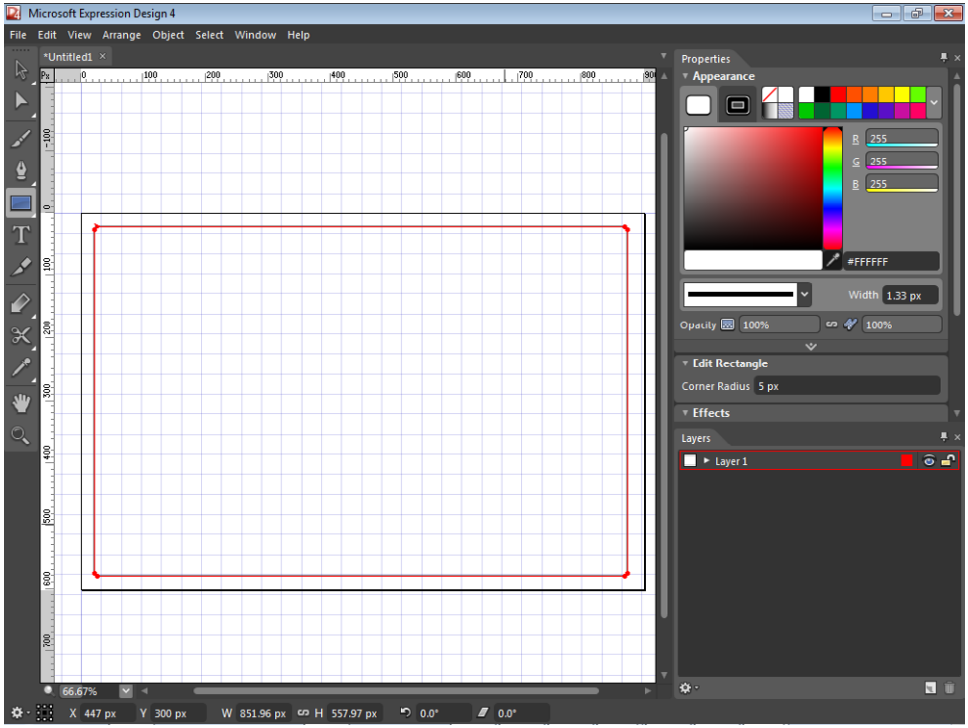
Tip If you don't want to draw your own template, you can open the finished file with Expression Design (Documents\Microsoft Press\Expression Web 4 SBS\Files\CH6-template .design), and then skip forward to step 38.

5. From the File menu, select New. In the New Document dialog box, type **Chapter6** in the Name field; **900 px** in the Width field and **600 px** in the Height field. Type **96 px/inch** in the Resolution field and then click OK.

6. From the View menu, select Show, and then enable Grid, Guides, and Edges.

   By showing the visual aids, you will have a better point of reference as you create your layout.

7. In the Toolbox, click the Rectangle tool, and then click and drag a rectangle onto the Artboard.

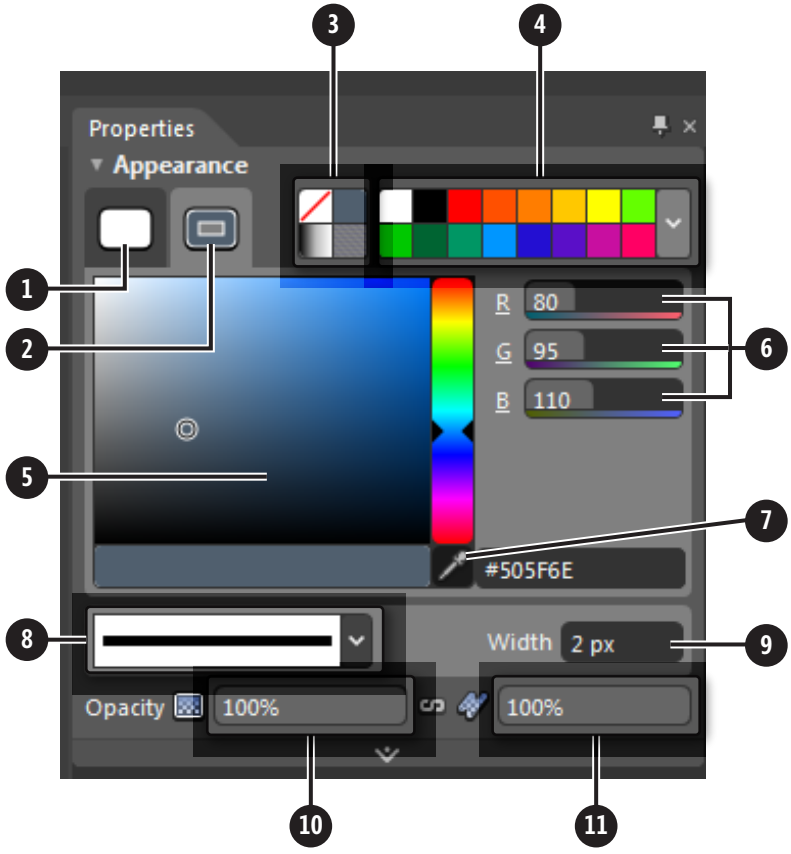


This simple rectangle will serve as the basic "container" for the Web page template. In the next few steps, you will add some visual enhancements to it.

8. Click the Selection tool at the top of the Toolbox, and then click the Artboard outside of the rectangle to hide its marquee so you can see the rectangle's appearance more clearly.

   This rectangle is a vector shape and you can manipulate all its properties in Expression Design.

   Using the Properties panel, you can adjust most of the shape's appearance properties The size of the shape at this point is not important. Just work on the shape as if it were the outside edge of the content area in a page and adjust it for balance and general appearance.



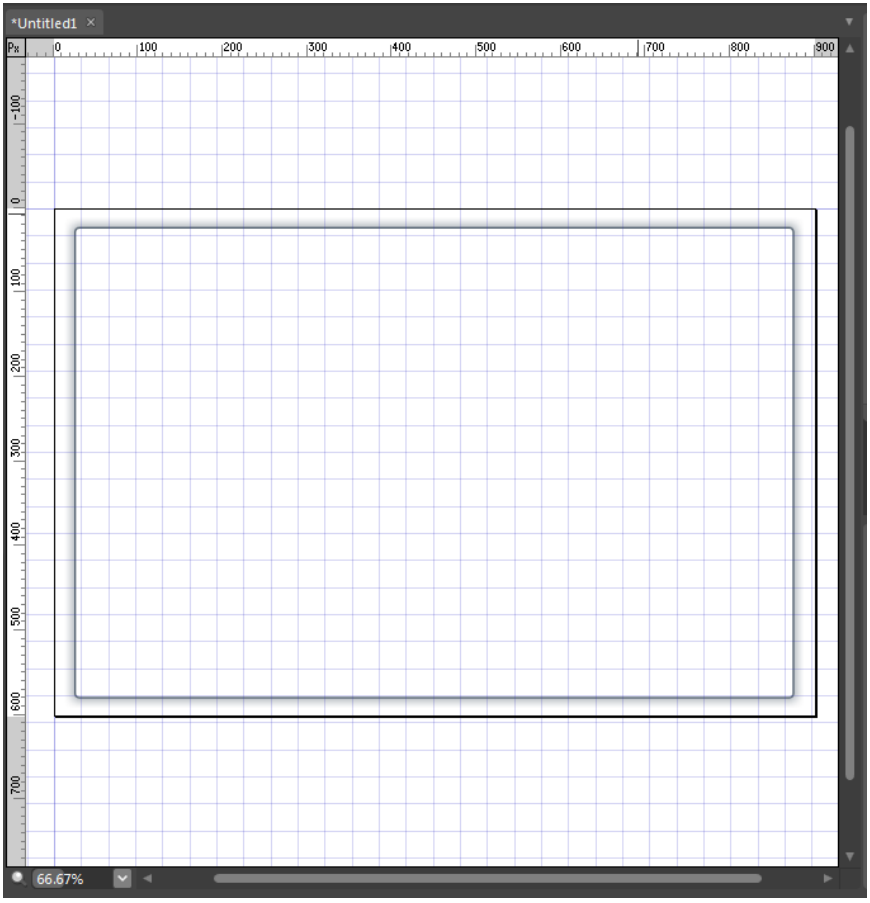
1. Fill button
2. Stroke button
3. Fill Type/Stroke Type
4. More Swatches button
5. Color Picker
6. Color Sliders
7. Color Dropper
8. Stroke Gallery
9. Stroke Width
10. Fill Opacity
11. Stroke Opacity

Tip Every shape has a "fill" and a "stroke." The tabs at the top of the appearance category of the Properties panel enable you to control the appearance of each, independently from the other.

9. Click the edge of the rectangle to select it, click the Fill button on the Appearance tab, and then click the white swatch. Click the Stroke button, click the drop-down arrow on the More Swatches button, and then click the #505F6E swatch in the Beach Swatches category.

10. Click the drop-down arrow beside the Stroke Gallery, click Basic Stroke, and then type **2px** in the Width field.

11. In the Effects Panel below the Appearance Panel, click the Add Effects button, point to Effects, and then click Outer Glow. Type **3** in the size field, **0.5** in the Opacity field, and use the Stroke color (#505F6E) for the Glow Color.



This is the basic shape for the page's content background. In the next few steps, you will create some of the common page areas.

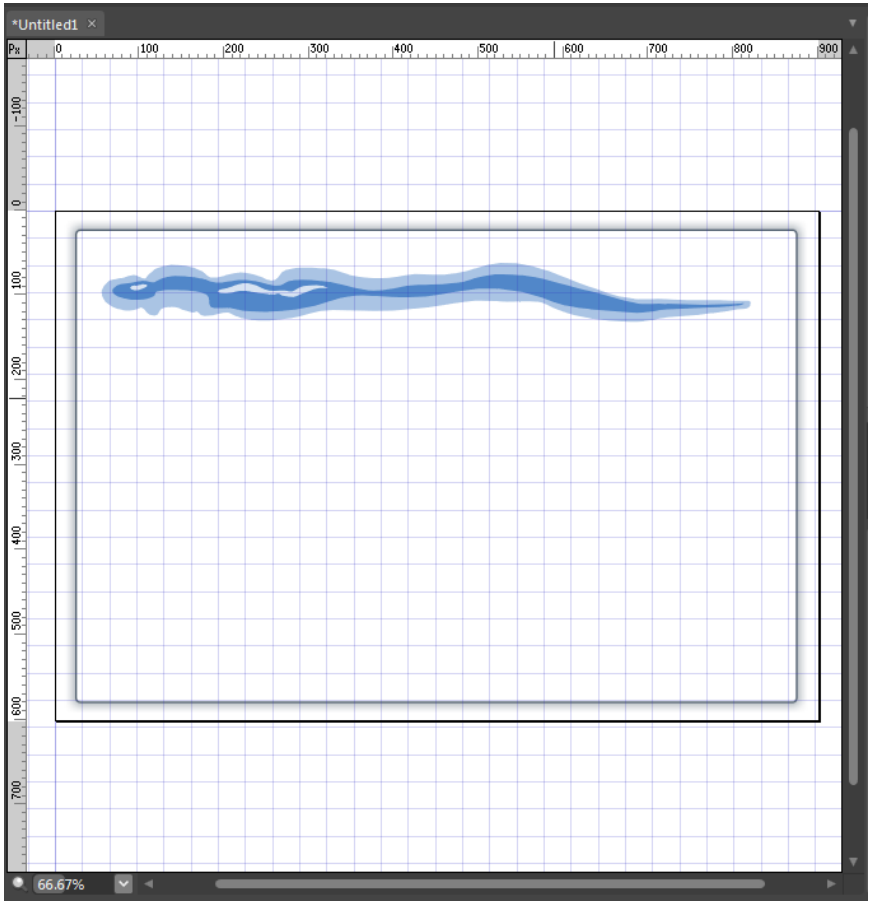
12. Using the action bar at the bottom of your workspace, set the Width to **850 px** and the X position to **450 px**, which centers the rectangle in the middle of the 900-pixel-wide document.

    Because this design will stretch to accommodate any height necessary to contain the Web page content, there is no reason to adjust the height of the shape.

13. Click the Paintbrush tool in the Toolbox, and then drag a horizontal line across the top of your shape to serve as the header area of the template.

14. Click the Stroke button, click the Stroke Gallery drop-down arrow, and then in the General group, click Thin Multi-layer. Set the width to **60px** in the Width field.

15. Click the More Swatches drop-down arrow. Click the #3C78C3 swatch in the Beach Swatches group, and then click the Selection tool in the Toolbox. Drag the anchor points on the selection to adjust the width and height of the brush stroke as necessary. Click the Artboard outside the objects so that you can see the results without the marquee guides.

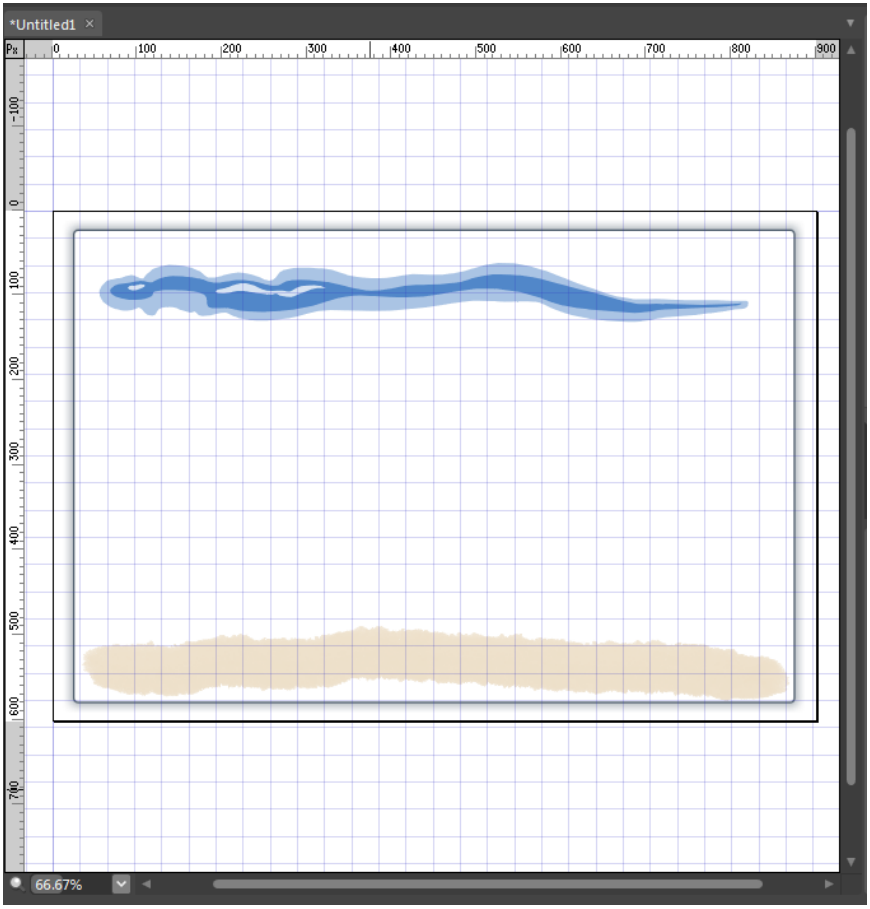


16. Click the Paintbrush tool again. This time, drag a line at the bottom of the rectangle shape where the footer would be.

Tip Drag in the opposite direction than you did for the header to get a more balanced appearance. The Paintbrush tool is designed to behave like a paintbrush, leaving a heavier mark at the beginning of a stroke and a lighter mark at the end.

17. Click the More Swatches drop-down arrow, and then click the #EBDCC3 color in the Beach Swatches category to give this stroke a sandy color. Click the Stroke Gallery drop-down arrow and then click the Fuzzy Thick Wet Brush in the Watercolor category.

18. Click the Selection tool in the Toolbox, and adjust the width and height of the stroke as necessary. As needed, click the Artboard outside of the objects so that you can get a clear view of your work so far.



At this point, you have a simple header and footer area; in the next few steps, you will add some additional strokes to help define other page areas.

19. Click the Pen tool, and draw a vertical line starting at approximately the 200 pixel mark from the left side of the Artboard. Click the Stroke Gallery drop-down arrow, and then click Fuzzy Brush in the Ink category.

20. Click the Stroke button at the top of the Appearance panel; click the drop-down arrow beside the More Color Swatches button, and then click #EBDCC3 in the Beach Swatches group.

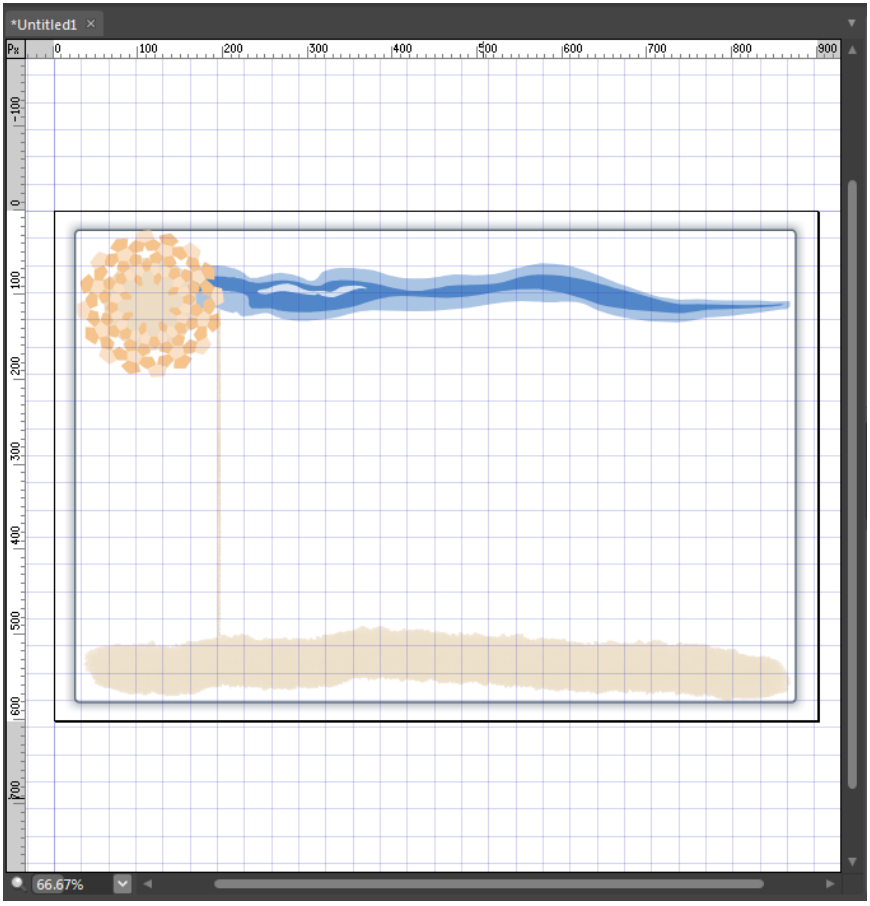
21. Click the Selection tool in the Toolbox and adjust the height of this line so that it's just touching the stroke at the top and bottom, but not overlapping them.

22. Click and hold the triangle in the lower-right corner of the Rectangle button, and then click the Ellipse button in the pop-up that appears. Drag a circular shape in the upper left of the header.

Tip When you're drawing a line and want it to be straight, hold the Shift key down as you draw. This technique works similarly with the Ellipse tool (to keep a shape circular) and with the Rectangle tool (to keep a shape square).

23. Click the Fill button in the Appearance panel, and then click #EBDCC3 in the Beach Swatches group. Click the Stroke button, and then click #F5C38C to give the shape a different outline. Click the Stroke Gallery drop-down arrow, and then click Geometric Flowers With Stars in the Design Elements category. Drag the Width slider beside the Stroke Gallery to adjust the size of the stroke.

24. Click the Selection tool again and adjust the shape to fit the overall drawing.



Now that you've drawn these page areas, the next steps are to create some color sample rectangles and slice the image, so that you can use it in your Web page.

25. Click the Text tool, and set your cursor on the left side of the blue brushstroke at the top of your document. Type **Summer Beach Template**.

    Notice that the text follows the center line of the brushstroke. If the text doesn't follow the brushstroke, then you've set the text cursor near—but not on—the centerline of the brushstroke.

26. Press Ctrl+A on your keyboard to select all the text you entered, then in the Text group of the Properties panel, select **Lucida Calligraphy, 36 pt**. Click the More Swatches drop-down arrow, and then click #967878 in the Beach Swatches category.

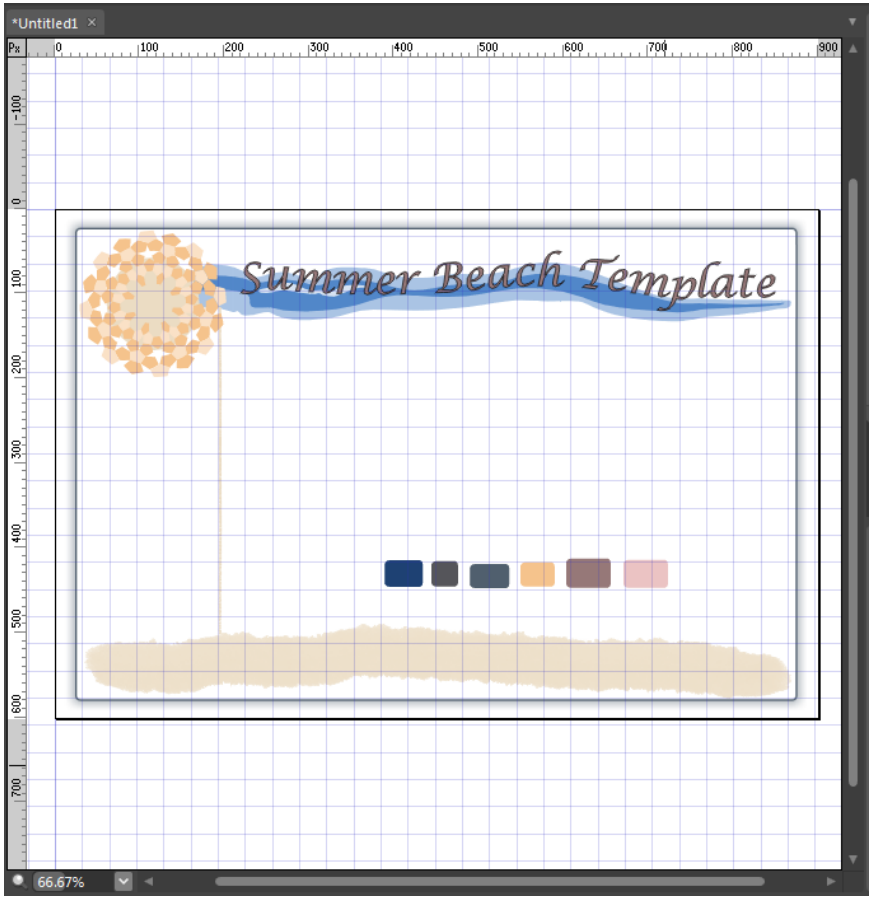
27. Click the Stroke button, and then click the black swatch to put an outline around the text, which will give it better separation from the background.

28. Click and hold the triangle in the lower right of the Ellipse button, and then click the Rectangle tool in the pop-up. Drag six small rectangles onto the document just above the footer brush stroke.

    Don't worry about uniformity or size and shape here. Think of these shapes as simple containers to hold the color samples that you will use in Expression Web to color your text, headers, links, and so on.

29. Click the Selection tool, and organize the rectangles so that they're out of the way of the other page elements. Click the first rectangle on the Artboard. Then, in the Beach Swatches category, click a color that you would like to use in your HTML template. Repeat this process to assign a different color from the Beach Swatches category to each of the six shapes.

Tip Make sure you've selected the Fill Color in the upper right corner of the window and not the Stroke.



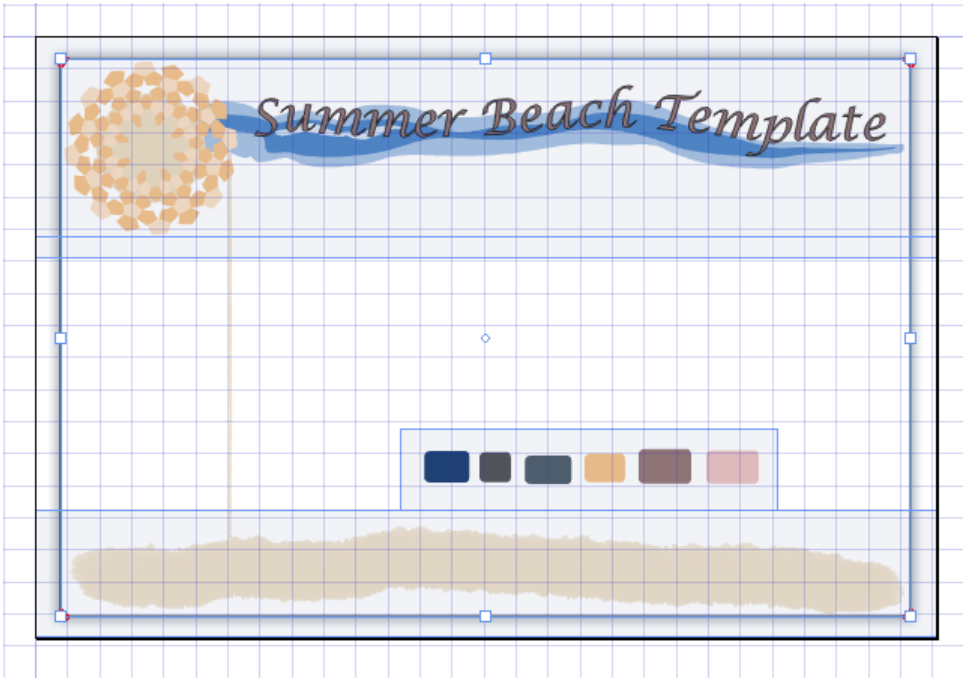
The colors that designers choose are up to them, based on individual preferences and the end goals of their design, but for this exercise, use these colors: **#1E4173**, **#55555A**, **#505F6E**, **#F5C38C**, **#967878**, and **#EBC3C3** (from left to right). These choices give you three cool colors and three warm colors to work with. By putting these colored rectangles on the Artboard, you can easily select the colors for your HTML and CSS as you build your page template in Expression Web.

30. Click the Slice tool in the Toolbox, and drag a slice from the upper-left corner of your document to the right side. Create a slice around the header graphics. The slice will be **900px** wide and the height will be such that it will contain the circular shape and the horizontal brushstroke.

Tip You can switch from the Slice tool to the Selection tool to adjust the size and shape of a slice, just as you can a shape.

31. Click and drag a slice at the bottom of the document to contain the footer graphics. This slice will be similar to the header slice but with a little less height.

32. Next, drag a slice across the width of your document just below the header slice. This slice should be **900px** wide, and about **20px** high. Finally, draw a slice around the color rectangles you added to the document.



Now that the slices are drawn, you will use the Properties panel to name them and specify their export file format.

33. Click the Selection tool, and then click the top slice on the document. In the Properties panel, click **JPEG** beside the Format field, and type **header** into the Slice Name field.

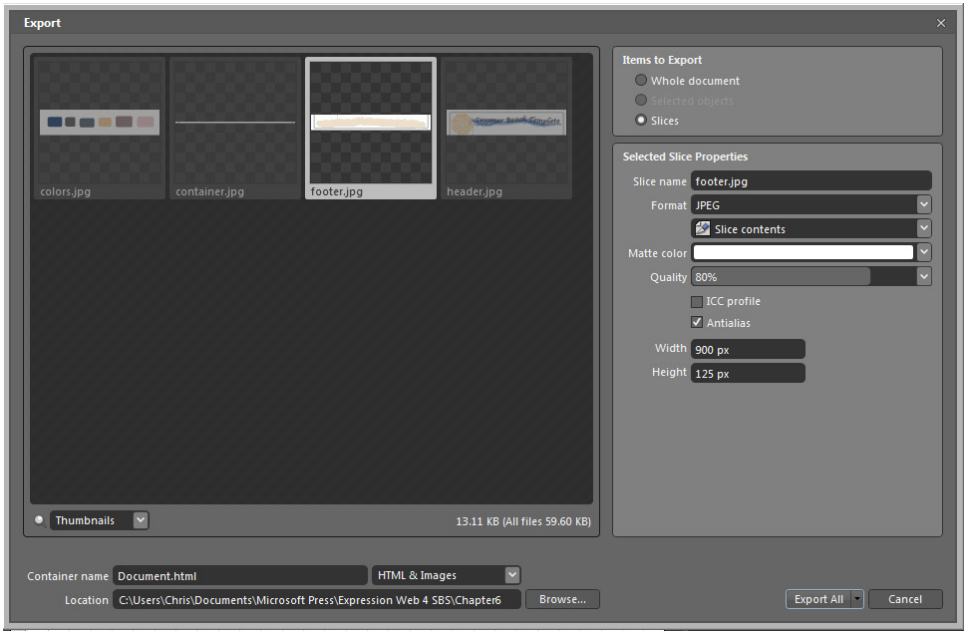
34. Click the next slice below the header slice. Click **JPEG** beside the Format field, and type **container** into the Slice Name field.

35. Click the slice that contains the colored rectangles, select **JPEG** beside the Format field, and type **colors** into the Slice Name field.

36. Click the bottom slice, select **JPEG** beside the Format field, and type **footer** into the Slice Name field.

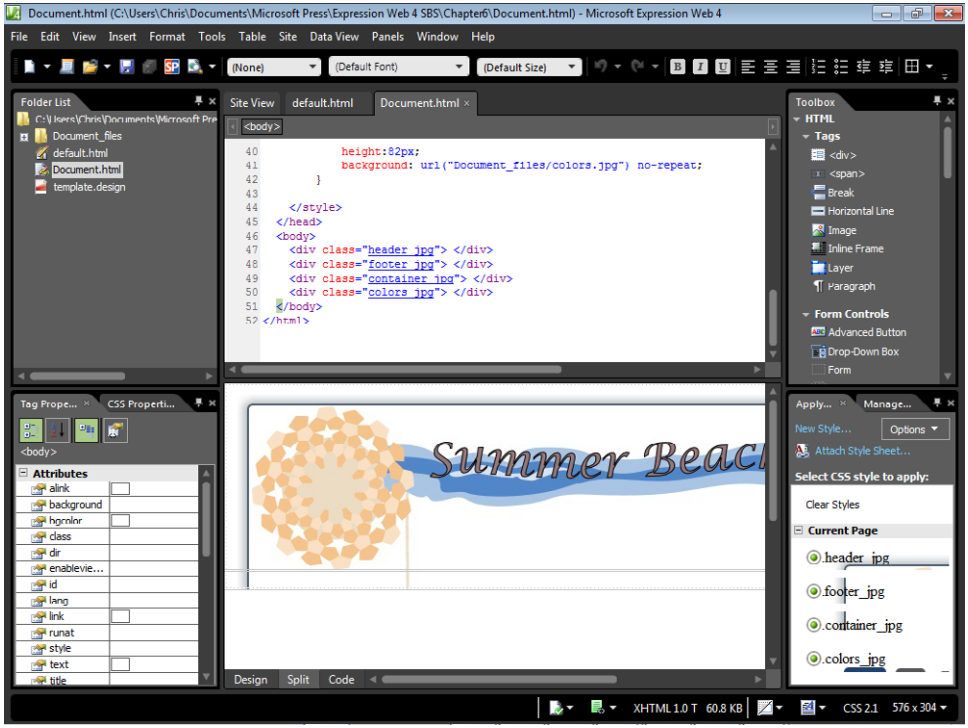
37. Click File, click Save, and then in the Save As dialog box, navigate to the One Page Site you made at the beginning of this section (Documents\Microsoft Press\Expression Web 4 SBS\Chapter6), type **template.design** into the File Name field, and then click Save.

38. Click File, click Export, and then in the Export dialog box, select Slices in the Items To Export field. Click the drop-down arrow beside the Container Name field, and select **HTML & Images**. Click the Browse button beside the Location field, and browse to Documents\Microsoft Press\Expression Web 4 SBS\Chapter6 in the Export Location dialog box. Then click Save.



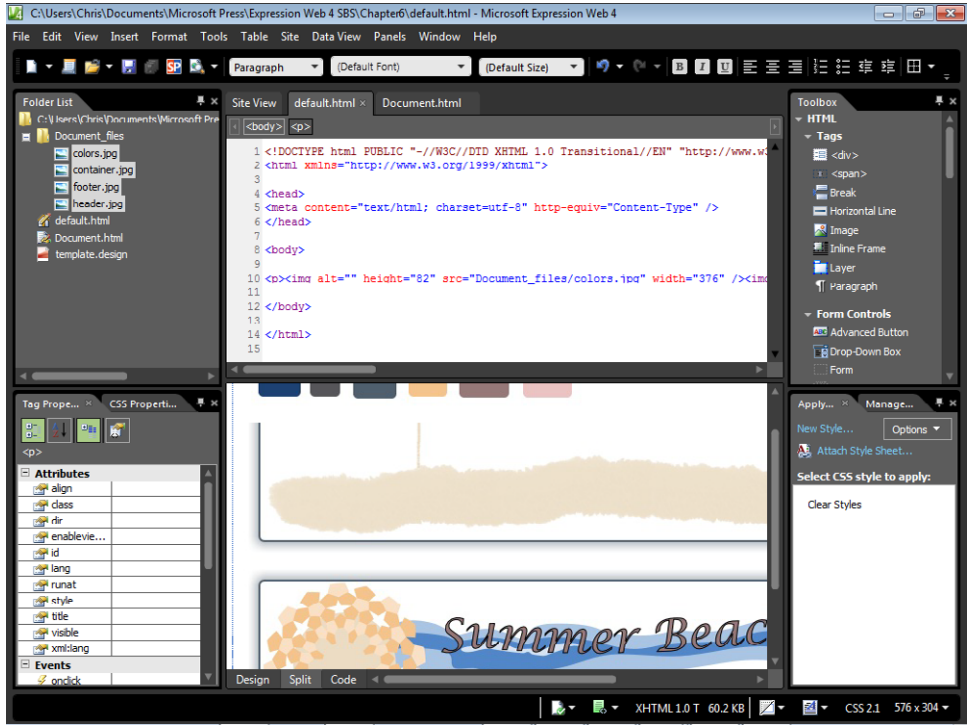
39. Click Export All in the Export dialog box. Expression Design *rasterizes* all the slices and exports a document (Document.html) and the images into the One Page Site you created at the beginning of this exercise.

40. Close Expression Design, return to Expression Web, select Recalculate Hyperlinks from the Tools menu, and then double-click Document.html in the Folder List panel to open it in the editing window.



Take a few minutes to examine the Design and Code panes of the Document.html file. You can see that Expression Design has created a page with a *div* for each of the slices that you created, and created CSS code to utilize the sliced images as the backgrounds of these areas.

41. Switch to the default.html file and expand the Document_files folder in the Folder List panel. Click colors.jpg, and then press the Shift key while clicking header.jpg to select all the files in that folder. Drag the files from the folder onto the Design pane of the default.html file. Click OK on each of the Accessibility Properties prompts, because you're just dropping the files onto the page for reference and not for an actual production use. Click Save on the Common toolbar to save the page.



By following these steps, you've just created a graphics package that you can use as building blocks for a Web page template. Although the work in Expression Design was purposefully simple, the level of detail that a designer can control is virtually unlimited. Consider gradients, multiple layers, live effects, and so on. For modern Web sites, the graphics tool is in many ways more pivotal to being able to create or modify a Web page template than the HTML editor.

If you liked using Expression Design, you might consider the book *Microsoft Expression Design Step by Step* as a learning resource. You'll also find more good tutorials and sample files for Expression Design on the Microsoft Web site.

Tip There are links to more Expression Design materials in the Sample Site's Chapter6.html file. If you need additional help with this chapter segment, check out *http://ExpressionWebStepByStep.com/Video*.

Note Click Exit on the File menu to close Expression Design. If you are continuing directly to the next exercise, leave Expression Web and the Chapter6.html file open. Otherwise, click Exit on the File menu to close Expression Web.

Creating an HTML Layout

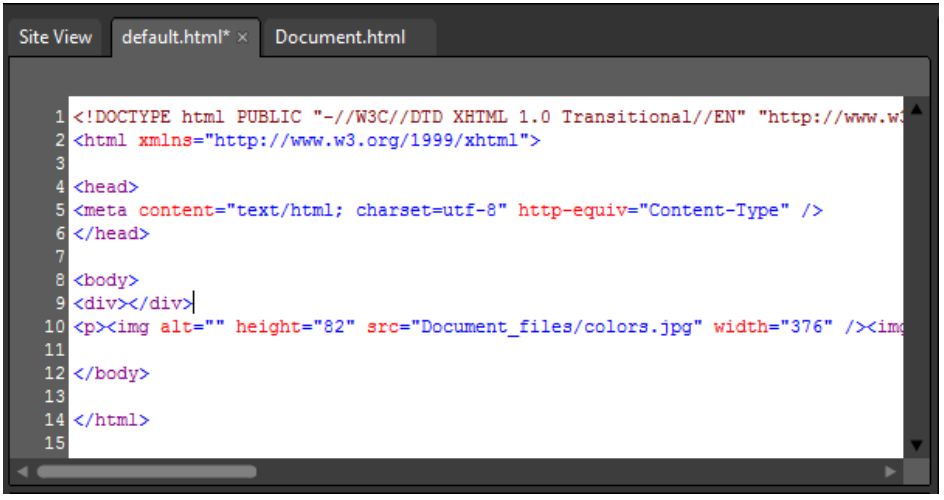
Now that you have the basic graphics layout created, sliced, and exported, you can go about the tasks required to implement these graphics in an HTML layout. Because of the steps in the previous chapter, such as naming the slices with meaningful names and slicing the graphics in specific sizes, the following steps will be much easier and require much less trial and error, and fewer revisions.

In this exercise, you will create your initial layout in an HTML page even though it will eventually be moved to a Dynamic Web Template (DWT). This is because you can easily preview an HTML page in a browser, but it's not so easy with a DWT. The cascading style sheet (CSS) you create will also be written in the HTML layout page itself, instead of an external style sheet. Eventually you will move the CSS from the HTML file to its own style sheet, but for speed and ease of use, it's just more efficient to work with a single file as you refine the HTML layout.

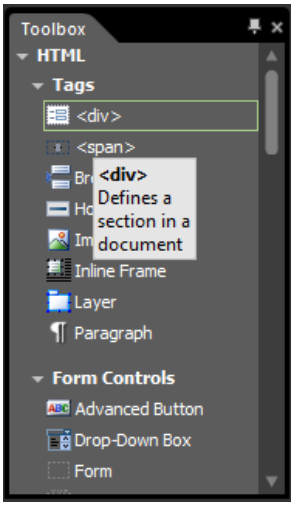
Create an HTML layout

Note Using the Chapter6 site that you created in the previous exercise, open the default.html file if it isn't still open from the previous exercise.

1. Click and drag the * * tag from the Toolbox on the upper-right side of the user interface to the Code pane just below the *<body>* tag of the default.html file.



The * * tag is the backbone of CSS/HTML layout. Its meaning is "division" and it's a block element that works as a container. With CSS, you can apply virtually limitless appearance modifications to it. This template you're using will contain 5 divs when you're done.

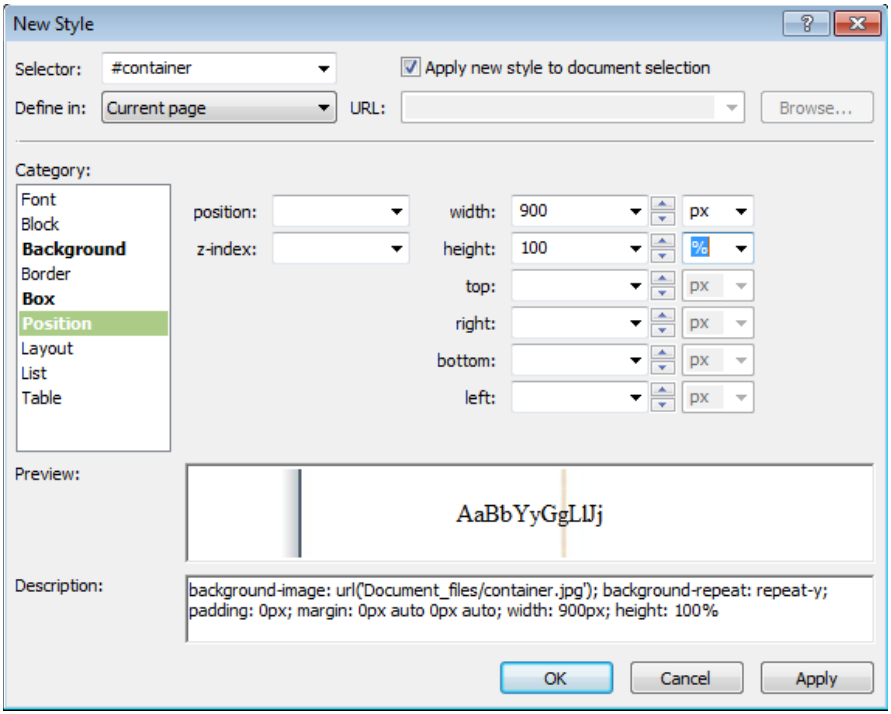


2. In the Design pane, hover your mouse pointer over the outside div, click the Block Selection label that appears so you can select the entire *div*, and then click New Style in the Apply Styles panel.
3. In the Selector field, enter **#container**. Select the Apply New Style To Document Selection check box, and set the Define In field to Current Page.
4. Click the Background category on the left side of the New Style dialog box. Click the Browse button beside the Background-Image field. Browse to and select container.jpg in the site's Document_files folder. Set the Background-Repeat field to **repeat-y**.

Tip X is always horizontal and Y is always vertical.

5. Click the Box category and set the padding to **0px** and leave its Same For All check box selected. Clear the Same For All check box beside the Margin field and set the following values: Top: **0px**, Right: **auto**, Bottom: **0px**, Left: **auto**.

   The margin-right and margin-left selections enable you to float the container in the middle of the screen. This is one of the most common questions with CSS layout and it's remarkably simple to do.
6. Click the Position category, and then set the width to **900px**, and the height to **100%**.



Notice that you can see all the styles in the Description field of the New Style dialog box.

7. Click OK in the New Style dialog box to set your changes.

   Although it might not look like much at this point, you've just set up the basic container for this entire template.

8. Drag another <*div*> from the Toolbox panel into the *div* that you just styled.

   Before you begin styling this *div*, you will need to know the height and width of the underlying image. Click the header image in the Design pane of your workspace. Look at the Code pane. You can see that it's 900 pixels wide and 200 pixels high.

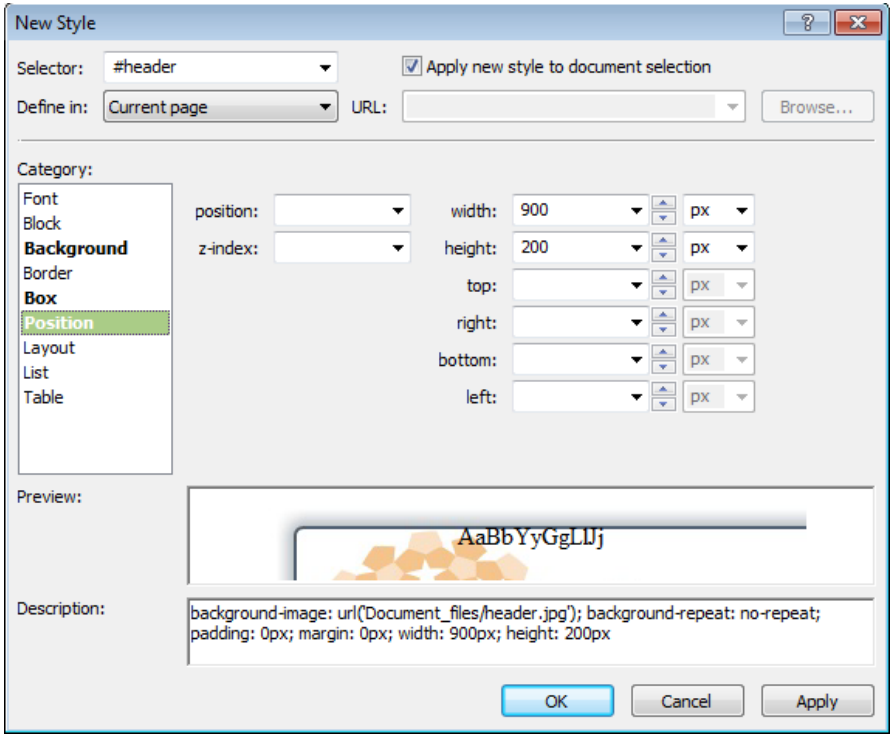
9. Select the *div* that you just added to the page by clicking its Block Selector tag in the Design pane, and then click New Style in the Apply Styles pane.

10. Enter **#header** in the Selector field, select the Apply New Style To Document Selection check box, and make sure that the Define In field is set to Current Page.

11. Click the Background category, click the Browse button beside the background-image field, and then browse to and select header.jpg in the site's Document_files folder. Set the Background-Repeat field to **no-repeat**.

12. Click the Box category and set padding and margin to **0px**. Leave both the padding and margin Same For All check boxes selected.

13. Click the Position category, and type **900px** in the Width field and **200px** in the Height field.



14. Click OK in the New Style dialog box to set your changes.

15. In the Design pane, set your cursor in the Header *div* you just styled, and then type **Header**.

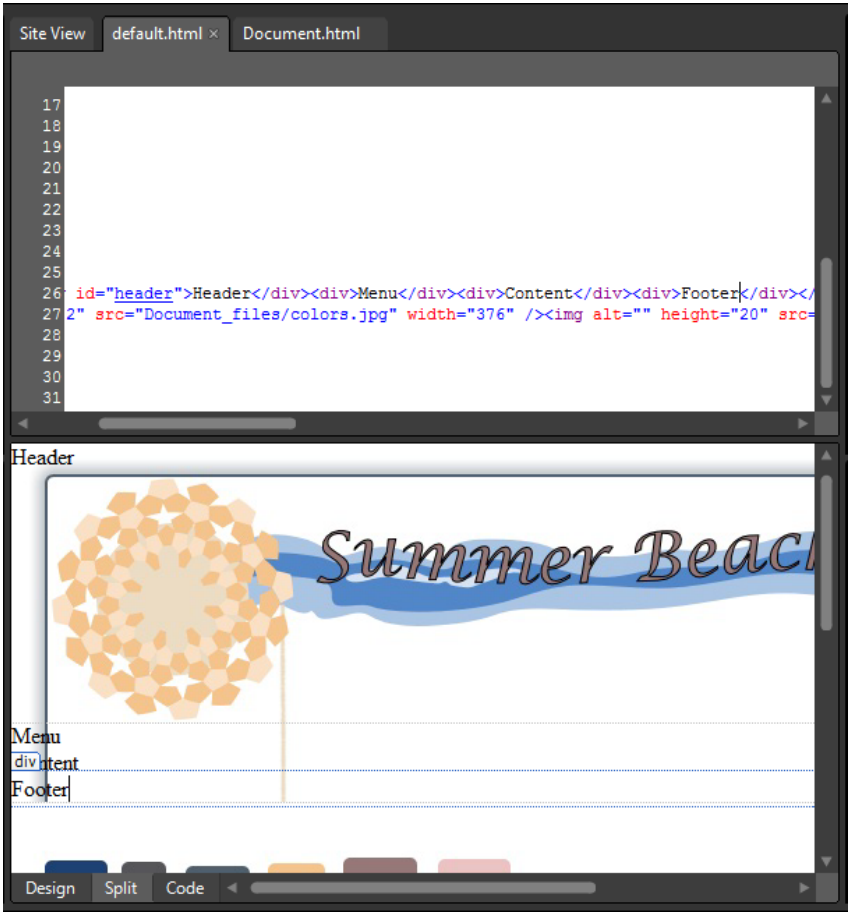
    This is recommended for two reasons: some browsers won't display *div* tags that have no content, and it provides good visual reference in the Design pane.

16. Drag another <*div*> from the Toolbox panel into the Code pane just after the closing </*div*> tag that you just typed the word "Header" into. Type **Menu** into this *div*.

Important It's imperative that you're careful in these steps. You want the *divs* that you're inserting to be "nested" inside the Container *div*, but not inside each other.

17. Drag another <*div*> from the Toolbox panel into the Code pane after the closing </*div*> tag that you typed the word "Menu" into. Type the word **Content** inside it.

18. Drag another <*div*> from the Toolbox panel into the Code pane after the closing </*div*> tag that you typed "Content" into, and then type **Footer** into this *div*. Click Save on the Common toolbar.



19. Right-click the Code pane, and from the Context menu, select Reformat HTML.

    This step will organize the Code pane for you and make working on this template easier.

20. Click the Block Selection tab in the Design pane for the Footer *div*, and then click New Style in the Apply Styles panel.

21. Enter **#footer** in the Selector field, select the Apply New Style To Document Selection check box, make sure the Define In field is set to Current Page, and then click the Background category.

22. Click the Browse button beside the Background-Image field, and then browse to and select footer.jpg in the site's Document_files folder. Set the Background-Repeat field to **no-repeat**.

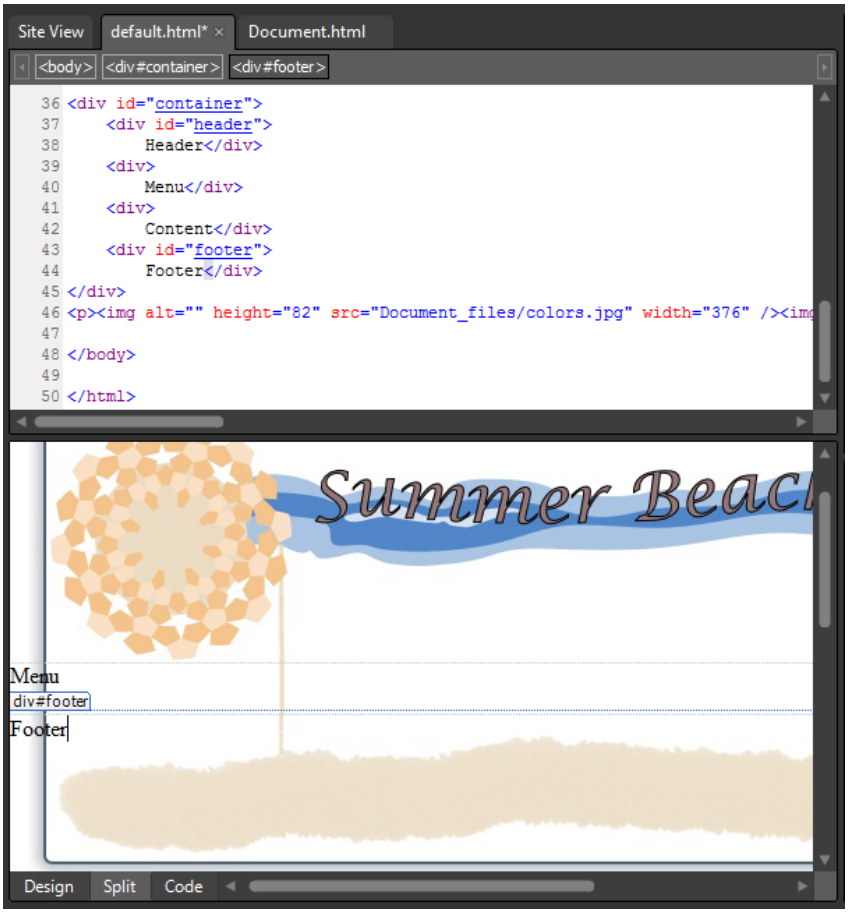
23. Click the Box category, set padding and margin to **0px**, and then leave both padding and margin Same For All check boxes checked.

24. Click the Position category, and type **900px** in the Width field and **125px** in the Height field.

25. Click the Layout category, and set the Clear field to **both**, and then click OK to set your changes and close the New Style dialog box.

    The *clear: both* style will ensure that the footer stays at the bottom of the design and won't be overlapped by either the content or menu divisions.

Tip The reason you were instructed to drag the images into the page is to serve as a reference while you're building these styles. Click the image in the Design pane and then look at the height and width attributes in the Code pane.



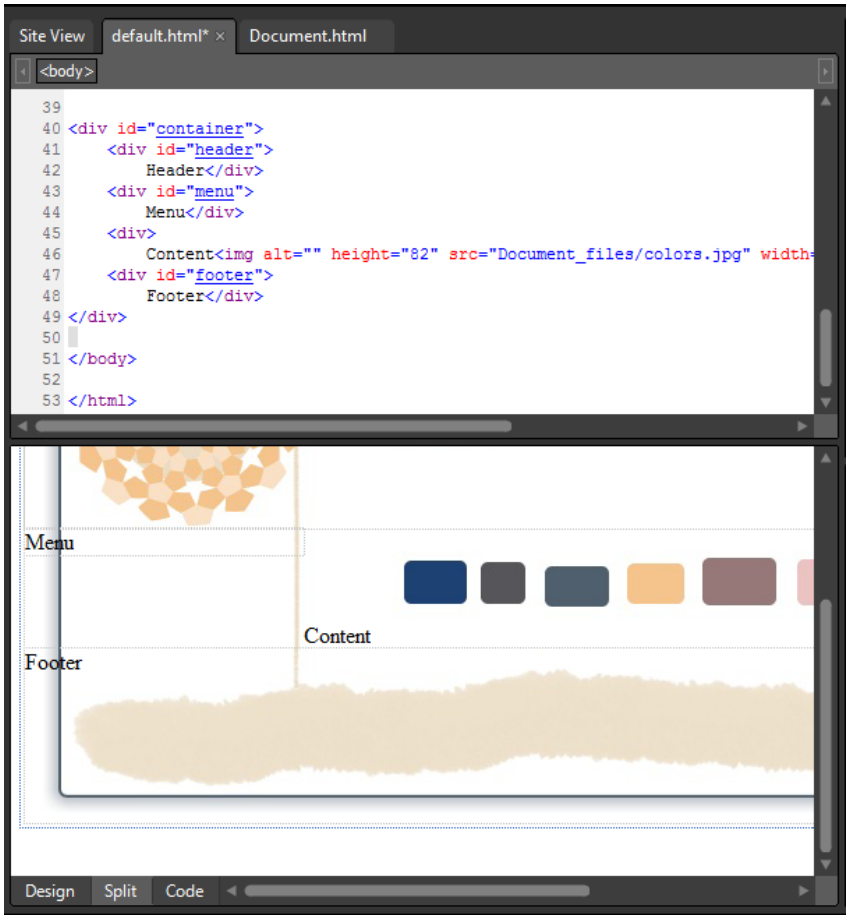
At this point, your HTML template is starting to take shape. Notice how the Menu and Content divisions are stacked horizontally. This is because the default appearance of a *div* is a *block element* 100% wide. In the next few steps, you will adjust these *divs* so that they line up beside each other vertically.

26. Click the Block Selection tag in the Design pane for the Menu *div*, and then click New Style in the Apply Styles panel.
27. Enter **#menu** in the Selector field, select the Apply New Style To Document Selection check box, and make sure Current Page is set in the Define In field.
28. Click the Position category, and type **200px** in the Width field.
29. Click the Layout category, set the Float field to **left,** and then click OK to set the style and close the dialog box.
30. Click the Block Selection tag in the Design pane for the Content *div*, and then click New Style in the Apply Styles panel.
31. Enter **#content** in the Selector field, select the Apply New Style To Document Selection check box, and make sure Current Page is set in the Define In field.
32. Click the Position category, and type **700px** in the Width field.
33. Click the Layout category and set the Float field to **right**, then click OK on the New Style dialog to apply your changes and close the dialog.

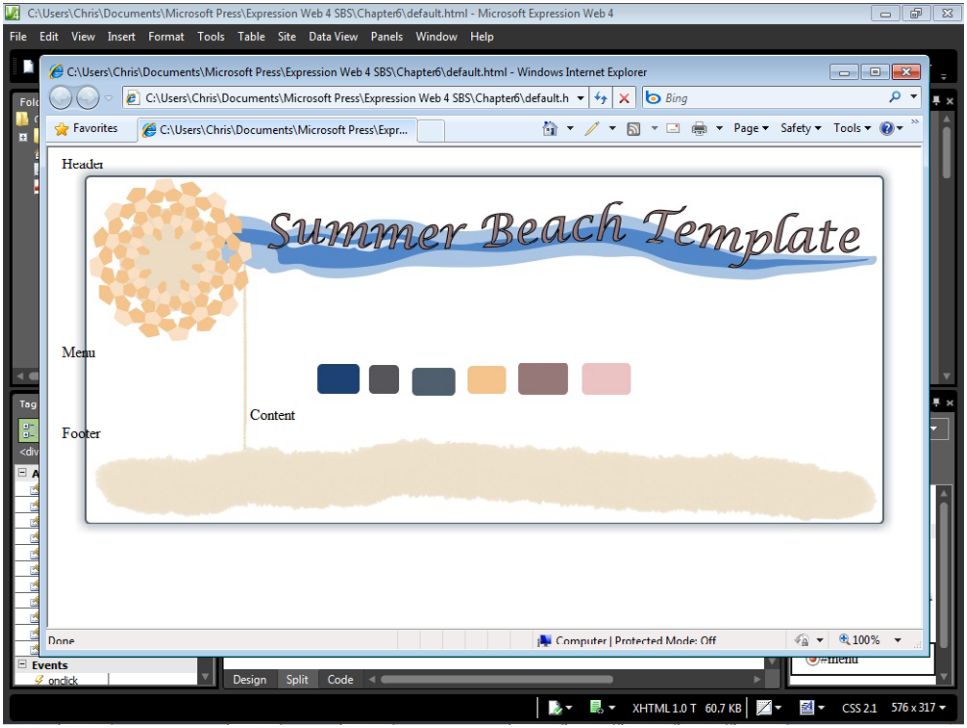
Tip If you designed the graphics template in Expression Design yourself, it's very likely that your image dimensions will be different. That's why you were instructed to insert the images into the page so you could have quick reference to their sizes by clicking them in the Design pane and looking at their sizes in the Code pane. Just substitute your image measurements where appropriate.

34. Click the colors.jpg image in the Design pane, and press Ctrl+X on your keyboard to cut it. Then set your cursor in the Content *div* just above the word "Content" and press Ctrl+V to paste it into the *div*.

35. Press the Ctrl key while clicking each of the three images below the Footer *div* tag, and press Delete on your keyboard to delete all of them. Then click the Block Selector tab for the <*p*> tag that contained them.



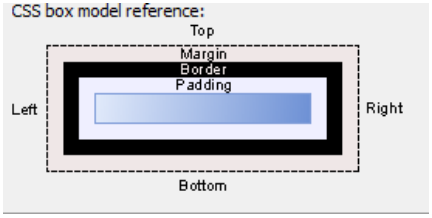
36. Click Save and then click Preview on the Common toolbar to check your work in a browser.



The browser rendering of this template is starting to look close to how it should. There are some details to take care of such as making sure that content inside the *divs* you created will be in the appropriate place, but the point that it's at right now is well past the halfway mark as far as layout is concerned.

37. Close the browser window and return to Expression Web.

The next steps are going to revolve around the CSS Box Model. You can think of a *div* as a box. If you're experienced in layout using tables, you can think of the *div* as a cell. What this template needs is some basic padding so that the content in each cell falls in the approximate location that it should. That will help transition into the next section where you will fine-tune the placement of content.

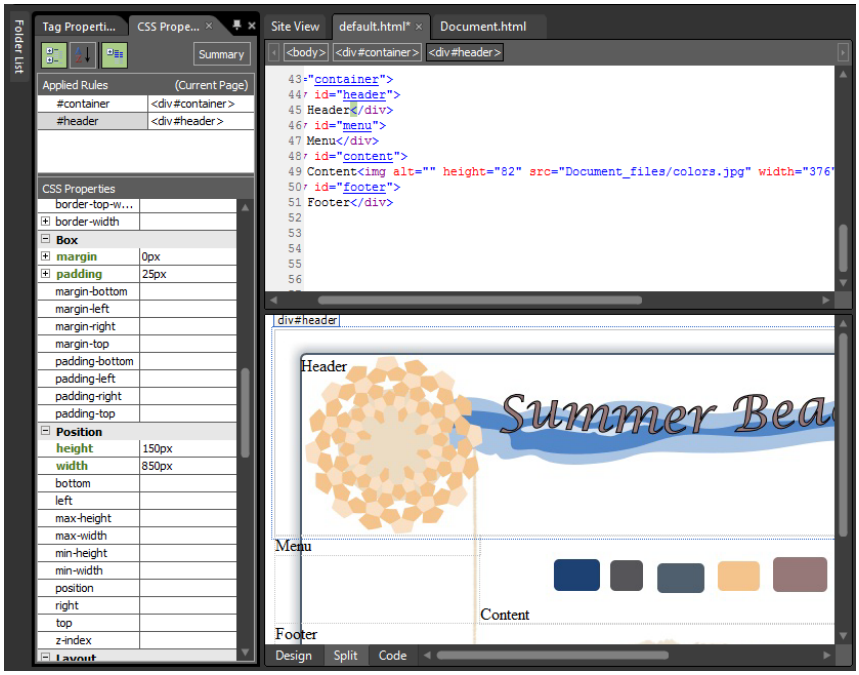


38. Click the CSS Properties tab on the lower left of the user interface to display the CSS Properties panel. Because the Folder List panel isn't going to do anything for you at this point, click the thumbtack icon on its upper-right corner to autohide it.

The CSS Properties panel is ideal for making quick modifications to existing styles, and you'll be using it to take care of the padding requirements for the layout *div*s that you've been working with.

Important Make sure you add the "px" in each of the following CSS specifications or you will get unpredictable results.

39. Set your cursor in the Design pane, Header *div*. Scroll the CSS Properties down to the Box section. Type **25px** in the Padding field. In the Position group, change the height entry to **150px** and the width entry to **850px**.



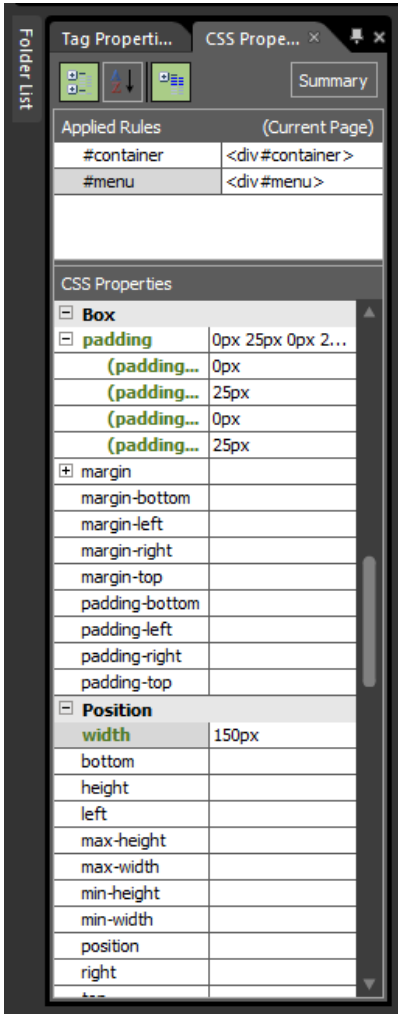
Because you've added 25px of padding to each side of the box model for the header, you have to reduce the width and height specifications to compensate for it.

Tip Notice that many of the categories in the CSS Properties panel have expandable entries, such as the padding entry you just worked on. If you make your entries in the fields that are exposed when the item is expanded, you will be creating shorthand CSS code. The entries below the expandable item will create longhand CSS code.

40. Set your cursor in the Menu *div* and scroll the CSS Properties panel down to the Box category again. In this *div*, we don't want padding at the top and the bottom, so click the plus (+) sign next to the padding entry, and enter these values: **0px** in Padding-Top, **25px** in Padding-Right, **0px** in Padding-Bottom, and **25px** in Padding-Left.

This will result in *shorthand CSS* code, which you will look at and compare to similar entries for another *div* in *longhand CSS*.

41. Change the width entry in the Position category to **150px** in order to compensate for the padding you added.

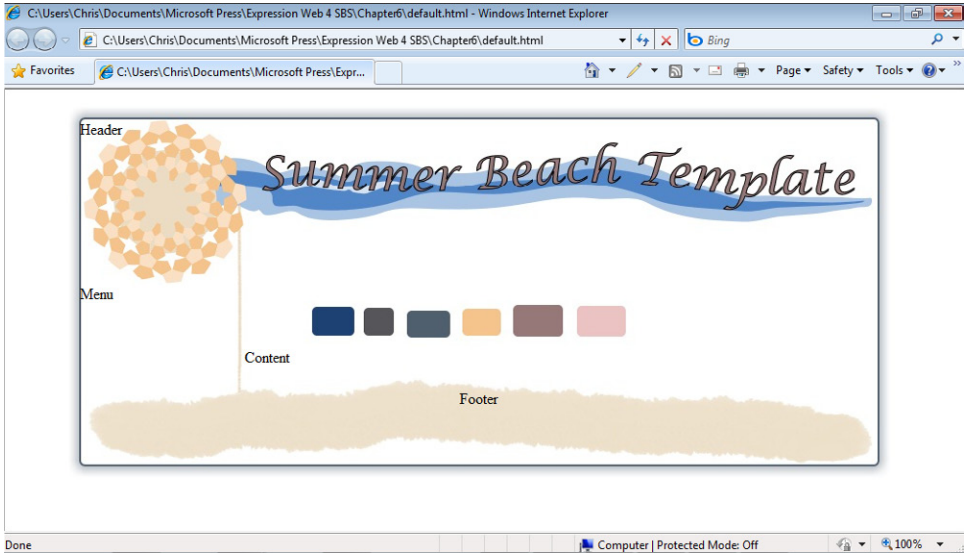


42. Set your cursor in the Content *div* and then scroll the CSS Properties panel down to the Box category. If the padding entry is expanded, click the minus (-) sign to close it and set the padding entries below it to: Padding-Bottom: **0px**, Padding-Left: **10px**, Padding-Right: **35px**, and Padding-Top: **0px**. Change the width entry under the Position category to **655px** to factor for the padding you just added.

43. Set your cursor in the Footer *div* and then scroll the CSS Properties panel down to the Box category. Enter **25px** in the Padding field and then click the plus (+) sign beside it. Notice how the 25px was added to all four padding fields.

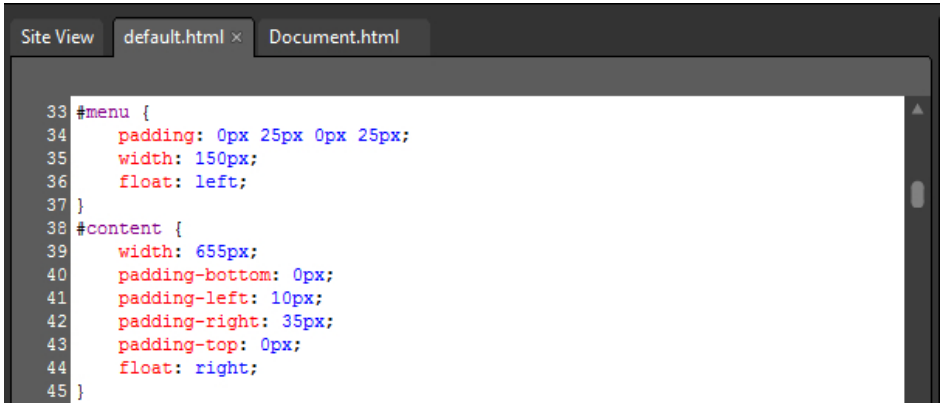
44. In the Position category, change the height to **75px** and the width to **850px**, and then in the Block category set the Text-Align field to **center**.

45. Click Save and then click Preview on the Common toolbar to check the results in a browser.



Just by adding some padding specifications to these page divisions, you now have text in the approximate place that it belongs. To further tune the placement, you will style the HTML tags that surround the content itself in the next section.

46. Close the browser and return to Expression Web. Scroll the Code pane to the top so that you can see the style block.



Notice that the *#menu* ID has its padding set in shorthand, and the *#content* ID has its padding set in longhand. This is a direct result of which fields you entered these specifications in using the CSS Properties panel. There isn't all that much difference in the end results when you choose longhand or shorthand CSS, but many users find it easier to read and modify longhand CSS.

Note If you are continuing directly to the next section, leave the Chapter6 site open; otherwise, you can close it.

Although this section took you through the tasks of laying out your graphics in an HTML template, it shouldn't be considered the end of what you need to learn about CSS/HTML layout. You will continue to learn about this topic through trial and error and research that you will conduct as you employ new methods to achieve the results you want. Consider that HTML 5 and CSS 3 are getting closer to release and general usage. As with everything in the Web design and development field, these techniques and standards are in a constant state of flux. The skills you learn by following examples in this book should give you enough familiarity with the concepts so that you can transition into higher-level techniques. There are a lot more books available on CSS than there are on Expression Web, so that should give you an indication of the breadth of knowledge out there on this topic.

Styling Major HTML Elements

In the previous segment, you used cascading style sheets code to style the divisions of an HTML template. Those exercises achieved the goal of getting the page layout arranged, but it also set up the ability to easily style the content that appears within the divisions.

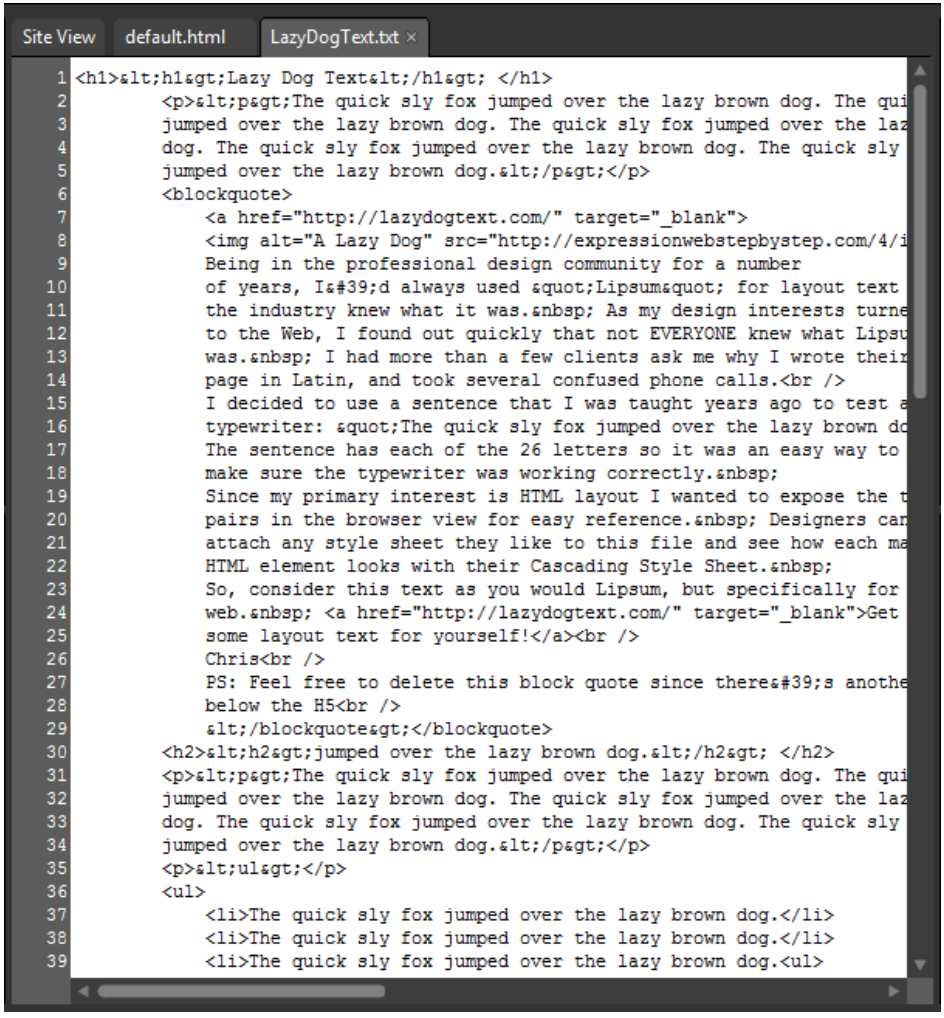
In this section, you will work on styling HTML tags in specific page divisions using cascading style sheets to refine the entire template and the appearance of content that will be included in it.

Using cascading style sheets in this way provides benefits, particularly with a growing site. For instance, all an author has to do is enter text in basic HTML tags, such as heading tags, paragraph tags, and lists. Because the designer set up the cascading style sheets for the template in a well thought out way, the content anyone enters will take on the appearance that the designer intended.

Create CSS styles for HTML markup

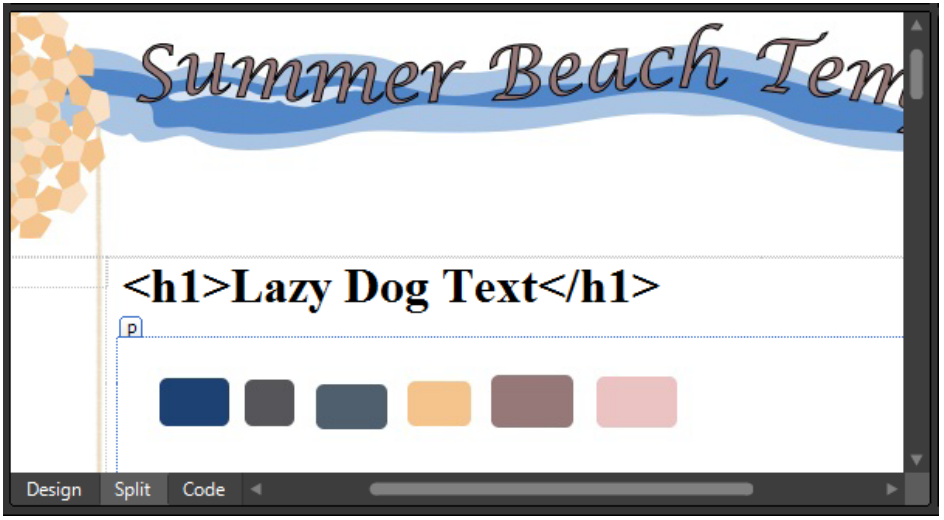
Note Using the Chapter6 site that you created in the previous exercise, open the default.html file if it isn't still open.

1. Click the Folder List tab on the upper left of the workspace that you set to autohide, and then click the thumbtack icon to lock it into place. Click the root folder in the Folder List, click File, click Import, and then select File.
2. In the Add File To Import List dialog box, browse to the Documents\Microsoft Press\Expression Web 4 SBS\Files folder, double-click LazyDogText.txt, and then click OK in the Import dialog box. The file is imported into the root of your site.
3. Double-click LazyDogText.txt in the Folder List panel to open it in the workspace.



Because you've opened a text file, with a .txt extension, Expression Web will open it in Code view. This text file has been designed to expose each of the 26 letters in the alphabet as well as a collection of the HTML tags most often used in Web page content. This file is helpful when designing CSS styles for content areas of a Web page.

4. Set your cursor in the text file and press Ctrl+A on your keyboard to select all the text. Then press Ctrl+C to copy all the code, and close the file.
5. Double-click the word "Content" in the Design pane, inside the content division of the default.html page, and then in the Code pane, right-click the highlighted word and select Paste from the context menu.
6. Scroll the Design pane to the bottom and click the colors.jpg image to select it. Press Ctrl+X on your keyboard to cut it and then scroll back to the top of the Design pane.
7. Set your cursor in the Design pane just after the *<h1>* text, press Enter on your keyboard to create a new paragraph, and then press Ctrl+V to paste the image into the paragraph.



By moving the colors.jpg image to the top of the page, it'll be easier to sample the colors for use in the CSS work that you're about to do. One of the first things you will accomplish is to get the text colored to match the graphics.

8. In the Apply Styles panel, click New Style, and type **body** into the Selector field. Leave the Apply New Style To Document Selection check box unselected, but make sure the Define In field is set to Current Page.

   What you're about to do is style the basic *<body>* tag using cascading style sheets. This is a very global selector, in that any text that isn't specifically styled will take on the properties you're about to set.

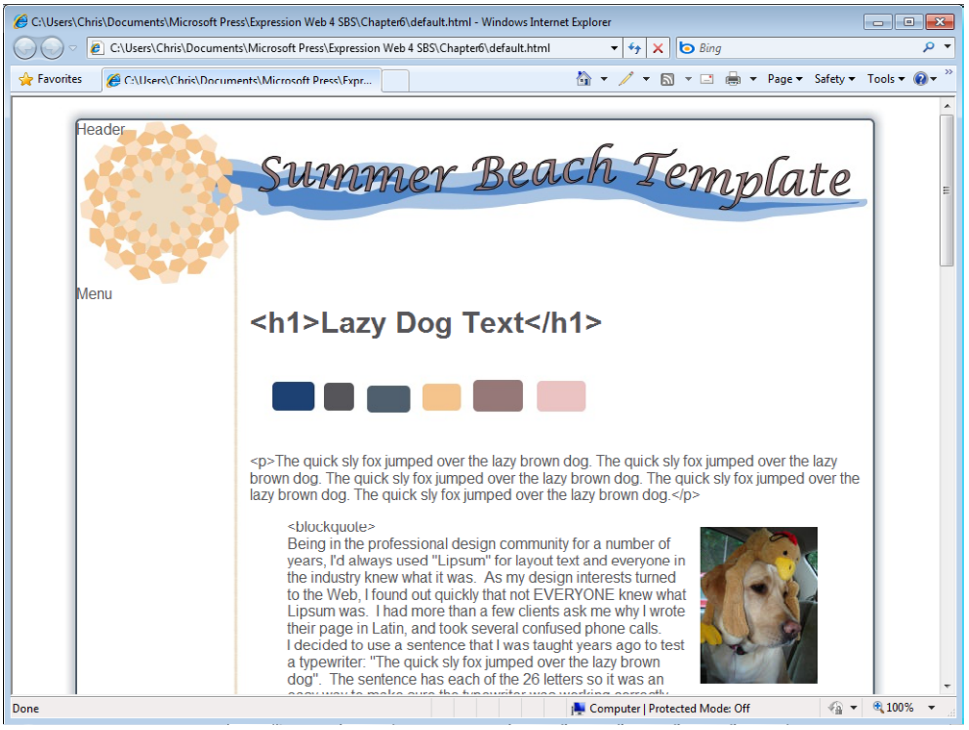
9. In the Font category, click the drop-down arrow beside Font-Family and then select Arial, Helvetica, Sans-Serif. Click the drop-down arrow beside the Color field, and then click More Colors. In the More Colors dialog box, click the Select button, then hold the color dropper over the grey color rectangle in the Design pane of your page, click it to set the color (*Hex={56,55,59}*), and then click OK.

10. Click the Background category and then click the drop-down arrow beside the Background-Color field. Click the white swatch (*#FFFFFF*).

    Although it might not seem like it's important to set the background to white, it actually is. If an element isn't styled, then it's up to the browser to decide how to present it. In this case, it could be very unattractive because the shadow and background of the graphics are white; the illusion would be ruined if the browser substituted a grey background.

11. Click the Box category. Leave the Same For All check boxes selected and enter **0px** in both Margin and Padding fields, and then click OK to set your changes and close the dialog box.

12. Click Save and then click Preview on the Common toolbar to check the page in a browser.



If you look closely at the browser view, you'll notice that there's less room between the header area of the page and the top of the browser viewport. Because there wasn't any specific margin and padding set for the body tag, the browser simply applied a default padding and margin setting. That's why specifically styling an HTML tag's properties gives you a high level of control. When a page has cross-browser differences, it's usually because of an unstyled HTML element that each browser chose to display your page with a different appearance.

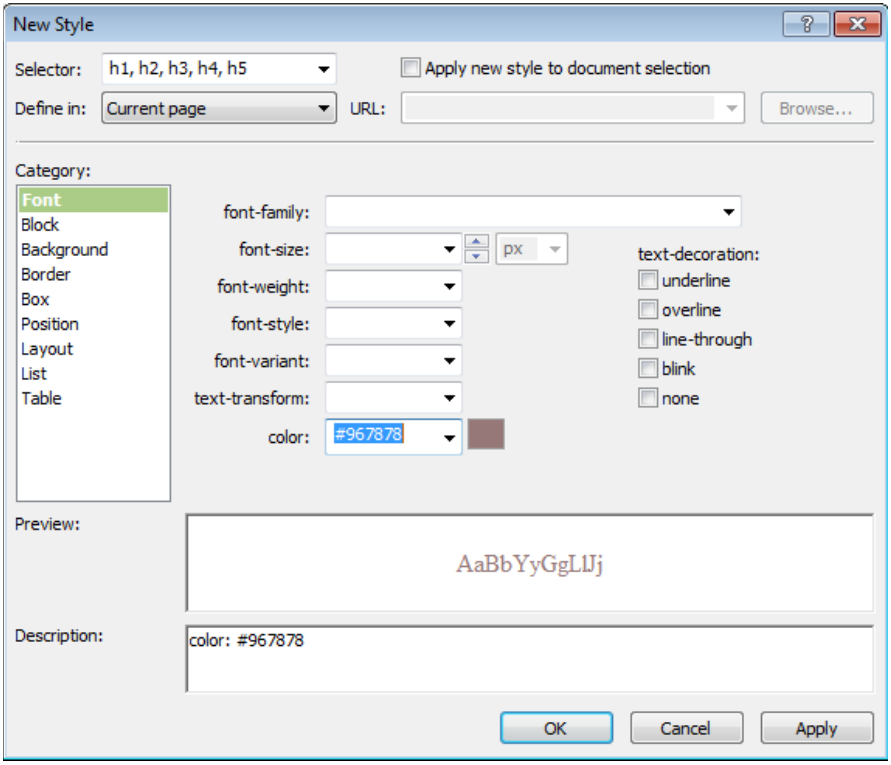
Also notice that all the text now appears in the grey color that you specified. Even though you didn't specifically set the text color for the content division, the parent container of everything on the page is the body tag. So, unless you specify a different font color for one of your divisions, it will appear in grey. By leveraging this inheritance property in CSS, the designer can achieve the maximum visual change with the minimum CSS code.

13. Close the browser and return to Expression Web.

    The next styling alteration you will make is to change the color of all the heading tags. You can do this easily by setting up a global style for all the heading tags and then, if needed, specifically styling the headings within certain page divisions.

14. In the Apply Style panel, click New Style. In the New Style dialog box, enter **h1, h2, h3, h4, h5** into the Selector field. Make sure the Apply New Style To Document Selection check box is not selected, and that the Define In field is set to Current Page.

15. Click the drop-down arrow beside the Color field and then click More Colors. Click the Select button, click the color dropper on the mauve color (*Hex={96,78,78}*) in the colors.jpg image in Design view, and then click OK in the More Colors dialog box.



This is a very simple style change. As you can see in the Description field, the only thing that has been added is a color specification. With this slight modification, every heading in the page will now have this color applied to it.

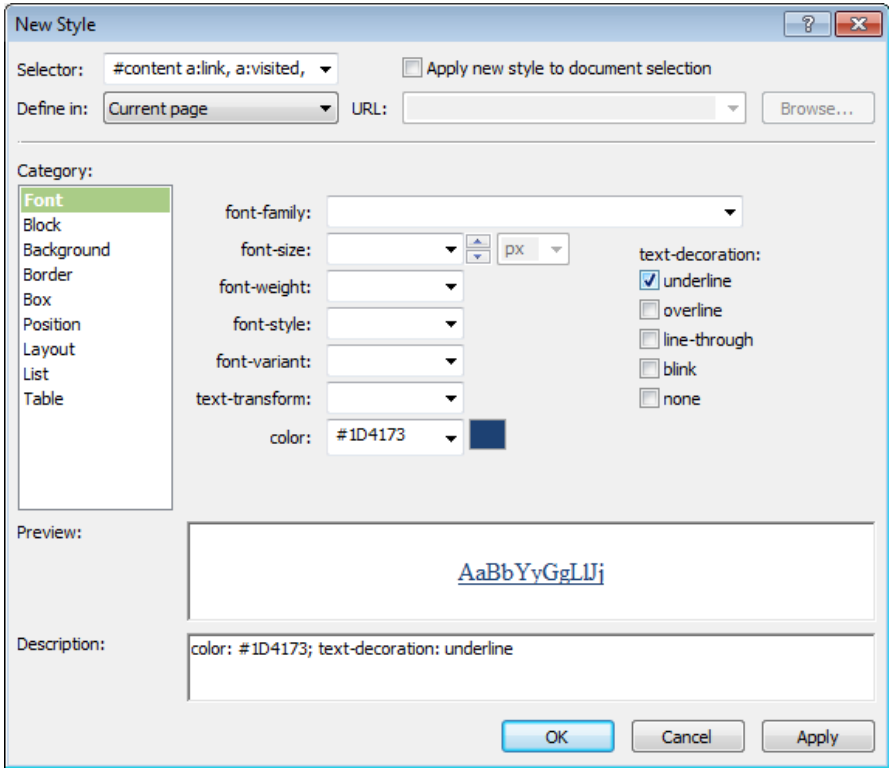
16. Click OK to set your change and close the dialog box.

17. Scroll through the content division in the Design pane of your page. Notice that everything in there is styled except for the links. In the next few steps, you will style those.

18. In the Apply Styles panel, click New Style, and in the Selector field, type **#content a:link, a:visited, a:active**. Make sure that the Apply New Style To Document Selection" check box is not selected, and that the Define In field is set to Current Page.

    In CSS, links are known as Anchor Pseudo-classes. There are four states to a link: default (*a:link*), visited (*a:visited*), active (*a:active*), and hover (*a:hover*). By setting Selector as you did, you've set up all the links which appear in the *#content division* of the page to take on the properties you're about to set. You can also set the style of all links using the *[a]* tag (without the state), which you will see later in this chapter.

19. Click the drop-down arrow beside the Color field and then click More Colors. Click the Select button, click the first blue rectangle in the Design pane view of colors.jpg (Hex={1D,41,73}), and then select the Underline check box, under the text-decoration field.



20. Click OK to set your changes and close the dialog box.

    Take a look at the Design pane of your page. You'll see that both links in this pane are now styled with the color you selected in the previous step. In the next step, you will use CSS to style what the link looks like when it's hovered.

21. In the Apply Styles panel, click New Style, and type **#content a:hover** into the Selector field. Make sure the Apply New Style To Document Selection check box is not selected, and that the Define In field is set to Current Page.

22. Click the drop-down arrow beside the Color field and then click More Colors. Click the Select button and then click the color dropper over the grey-blue (Hex={4F,5F,6E}) rectangle in the Design pane view of colors.jpg. Click OK in the More Colors dialog box.

23. Select the None check box beneath the Text-Decoration field and then click OK to set your changes and close the dialog box.

24. Click Save and then click Preview on the Common toolbar to check your page in a browser.

    Take a few minutes to examine the page and most importantly, the newly styled links in the content area. Make sure that the hover property is working as expected. Styling links is an easy way to provide some interactive feedback to the user. In a content area of a page, you should keep in mind that you don't want to style the link to such an extent that the user doesn't realize that it's a "clickable" link. This isn't a problem in menu areas where the user will assume that your highly styled links are navigation elements.

25. Close the browser window and return to Expression Web.

    In the next steps, you will deal with the header division in the page. The end goal of this work will be to create an *h1* that will be below the site title image, and as a bonus, you will learn a technique to include invisible text. The reason for this is simple: the site title is part of the graphics; therefore, it can't be read by screen readers or search engines. We could never achieve this text appearance with plain text so we've used a graphic. To counter this situation, we will use text that will be invisible to a visitor, yet present for search engines and screen readers.

26. Set your cursor in the header division and select the text "Header". Double-click the <*span*> tag in the Toolbox panel on the right side of the editing window. The word "Header" is replaced by <*span*></*span*>. Set your cursor in this span and type **Summer Beach Template**.

A <*span*> is very similar to a <*div*> with the main difference that a division is a block element by default, and a span is an inline block by default. What that means is you can have a number of spans next to each other horizontally without any modification of the default styles like you have to do with a <*div*>.

27. Click the <*span*> tag on the Quick Tag Selector to select the entire tag and all its contents. Then click New Style on the Apply Styles panel.

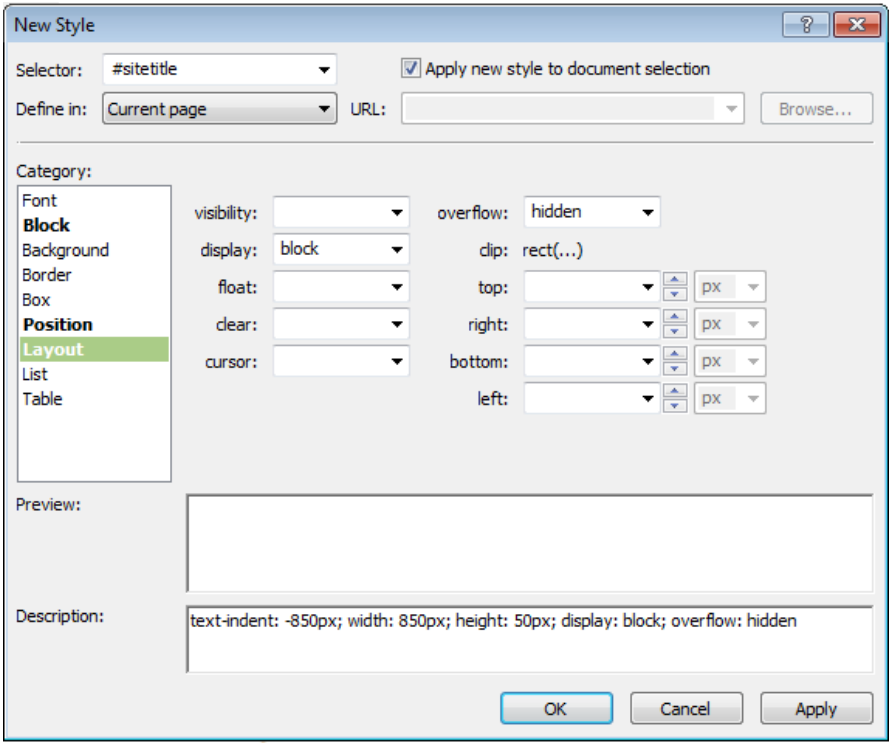
28. Type **#sitetitle** into the Selector field, select the Apply New Style To Document Selection check box, and make sure the Define In field is set to Current Page.

29. Click the Block category and enter **-850px** in the Text-Indent field.

    This causes the text to be negatively indented by 850px.

30. Click the Position category and enter **850px** in the Width field and **50px** in the Height field.

31. Click the Layout category, set the Overflow field to **hidden**, and then set the Display field to **block**.

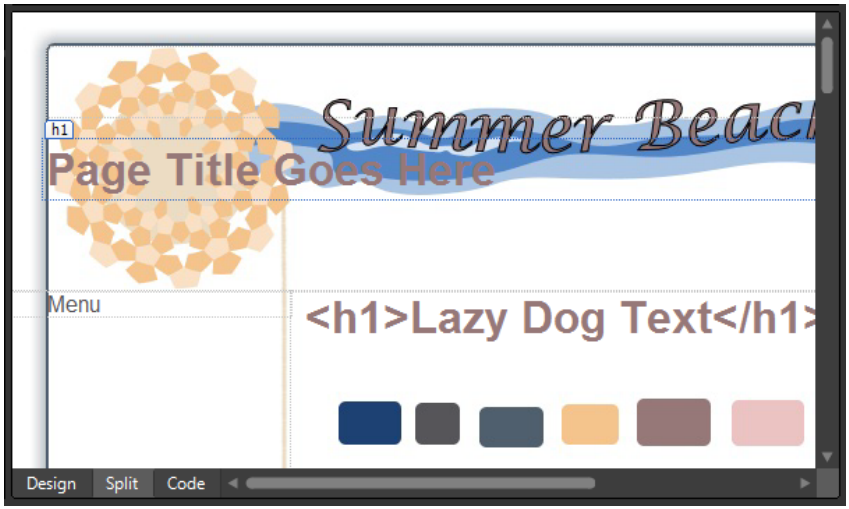


32. Click OK to set your changes and close the dialog box.

    Notice in the Design pane of your page that you clearly have a span, but the text isn't visible. This result could also be accomplished by using the display property of "none" or "hidden," but these techniques can be misinterpreted by search engines and your site could be penalized in search results for it.

33. Set your cursor in the Code pane just after the closing ** tag and type **Page Title Goes Here**.

34. Select this text in the Design pane, click the drop-down arrow on the Styles box of the Common toolbar, and then click Heading 1 *<h1>*.



The first <*h1*> in your page is a key to good search engine results. Having an <*h1*>, page title, and URL all matching the meaning of the page content is virtually unbeatable for search engine positioning. Fortunately, CSS can enable you to make this most important heading look like whatever you need it to in order to fit your overall design.

35. Select New Style in the Apply Styles panel. Enter *#header h1* in the Selector field and make sure that the Define In field is set to Current Page.

36. Click the Box category and enter **0px** in the top padding field. Leave the padding Same For All check box selected.

37. Clear the Same For All check box for the margin field and enter **75px** in the Top field, **0px** in the Right field, **0px** in the Bottom field, and **185px** in the Left field. Click OK to set your changes and close the dialog box.

Those simple margin changes have moved the *<h1>* that appears in the Header division to an appropriate position.

38. In the Design pane, double-click the text "Footer" in the footer division of the page to select it and then type **Top**.

    In the next few steps, you will create a bookmark link to the top of the page, add some text to the footer, and then fine tune the presentation.

39. Select the word "Top" and then right-click it. In the context menu, choose Hyperlink. In the Insert Hyperlink dialog box, type **#header** into the Address field, and then click OK.

    This is a nice little bonus of using CSS IDs on various page elements. You don't need to set up bookmarks and their respective hyperlinks as you would have in the past. Any link to an ID element will behave as a bookmark. So, when you click this link in a browser, the page will automatically reposition to the top.

40. Set your cursor in the Design pane, just after the link you inserted, and then press Shift+Enter on your keyboard to create a line break. Type **Copyright 2010 Summer Beach Template** and then press Shift+Enter on your keyboard again to break to a new line. Type **Terms of Use | Privacy | Contact**.

The footer content is beginning to take shape, but it needs a little work to style it appropriately to match the rest of the template.

41. In the Apply Styles panel, point to *#footer*, click the drop-down arrow on it, and then click Modify Style.

42. In the Modify Styles dialog box, enter **12px** in the Font-Size field, **600** in the Font-Weight field, and **#FFFFFF** (white) in the Color field. Click OK to set your changes and close the dialog box.

43. Click New Style on the Apply Styles panel and type **#footer a** into the Selector field. Make sure that the Apply New Style To Document Selection check box is clear and that the Define In field is set to Current Page.

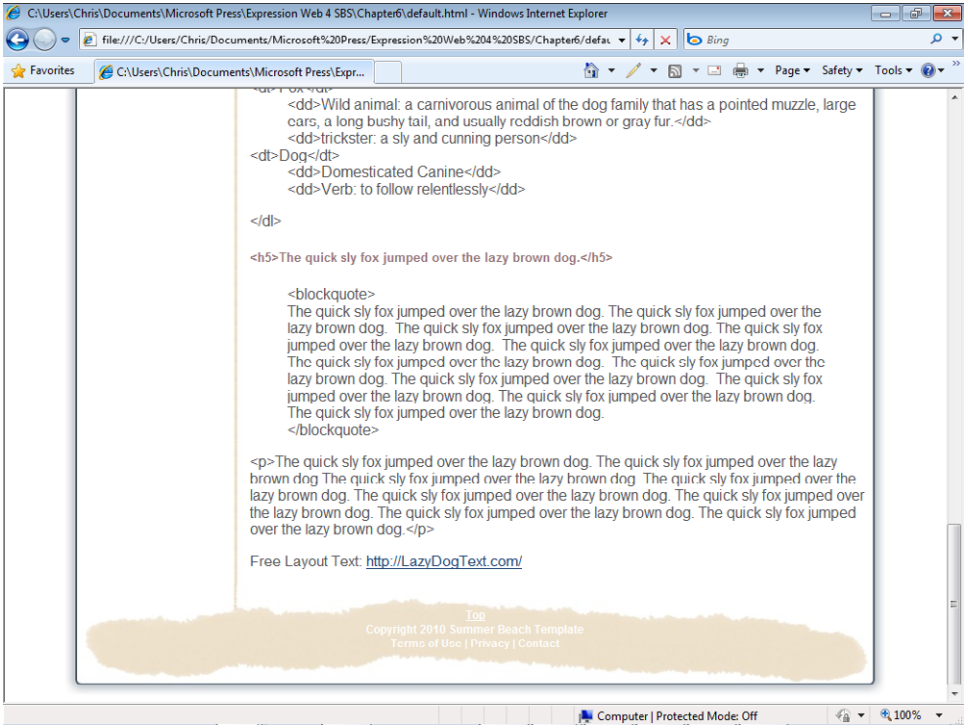
44. Set the color field to #FFFFFF (white) and then click OK to set the change and close the dialog box.

That minor change to the anchor pseudo classes has now matched the current, and any future hyperlinks in the footer, to the rest of the footer text.

45. Scroll the Design pane so that the colors.jpg image is visible, because you'll need to sample the grey color out of it. Click New Style on the Apply Styles panel, and then type **#footer a:hover** into the Selector field. Make sure the Apply New Style To Document Selection check box is not selected, and that the Define In field is set to Current Page.

46 Click the drop-down arrow beside the Color field and then click More Colors. In the More Colors dialog box, click the Select button, and then click the color dropper on the grey color swatch (Hex={56,55,59}) in the colors.jpg image in the Design pane, and then click OK. Click OK in the New Style dialog box to set your changes and close the dialog box.

47. Right-click the Code pane of the page and then in the context menu, choose Reformat HTML. Click Save and then click Preview on the Common toolbar to check your page in a browser. Scroll the page to the bottom and check your footer division and make sure that it appears as expected. Check the hover state of the link you styled and make sure that it does refocus the page to the header division.



Review the visual presentation of your page. Think about what else you might want to do as a designer, such as setting up CSS styles for the <*blockquote*> or other HTML tags, perhaps adding some style to the HTML list elements, and so on. The choices are endless, and you should feel free to experiment as much as you like.

Although you didn't work on the Menu page division, that's only because the site architecture and navigation haven't been established yet. You will complete those tasks in the next section.

This exercise has been fairly significant, whether it's apparent to you or not. You've styled an HTML template based on a custom graphics composition in a very logical and efficient way, and all the while maintained the appearance you wanted without sacrificing semantic value or HTML validity. You were also able to do this significant work with user-friendly New Style and Modify Style dialog boxes as opposed to manually writing CSS code.

Note If you're not continuing directly to the next exercise, you can close the default.html file and then exit Expression Web.

Designing the Site Architecture and Navigation

The folder structure, page organization, and navigation of a site go hand in hand. Besides keeping the site "clean" and logically organized, the designer has an opportunity to improve usability by designing the basic folder structure of the site. By naming the folders intuitively and pertinently to the information their pages contain, they can actually attain higher search engine results and make it easier for people to share links to specific pages or sections of the site.

In this section, you will arrange the structure of the site, create a Dynamic Web Template to ease the creation, deployment, and maintenance, and then add navigation to tie it all together.

Design the folder structure and navigation for a site

Note Using the Chapter6 site that you created earlier, open the default.html file if it isn't still open from the previous exercise.

1. Double-click the word "Menu" inside the menu page division and type **Home**. Click the bullets button on the Common toolbar to change this text to a list item in an unordered list.

An unordered list is ideal for a navigation menu. What is a navigation menu at its most basic other than a list of links? By creating this list, you'll be able to get an understanding of the coming folder structure, plus when the structure is complete, these list items can be converted to links and styled heavily with CSS.

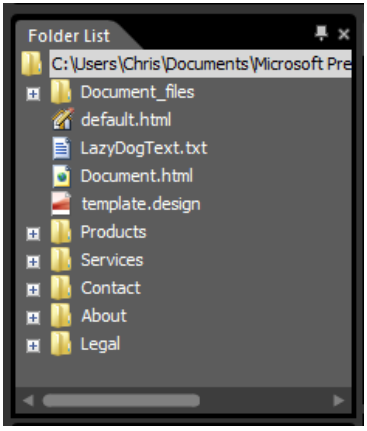
2. Press Enter on your keyboard after each term, and then type the next term so that you create a list containing **Products**, **Services**, **About Us**, and **Contact Us**. Press the Enter key twice more after the last list item to close the list.

3. Click Save on the Common toolbar to save your changes to the page.

Besides being the beginning of your site navigation, this list also gives you some direction as to how you will need to design the folder structure of the site.

4. Click the root folder in the Folder List panel, click the drop-down arrow beside the New Document button, and then click Folder. Type **Products** as the folder name.

5. Repeat the folder creation steps to create four more folders in the root of your site named **Services**, **About**, **Contact**, and **Legal**.



You added the folder "Legal" to the site so that you'd have a place to keep the Terms of Use and Privacy page that you'll link to on the footer.

6. Right-click the Document_files folder in the Folder List panel and then choose Rename from the context menu. Rename the folder to **Images**.

   When you rename a folder in Expression Web, all the references to files within it are automatically updated. Although it's not a necessity to rename this folder, it's always a good practice to keep folders and files named in a way that will mean something to you and anyone who might need to work on the site in the future. As the number of files and folders in a site becomes greater, the value of this practice becomes more and more important.

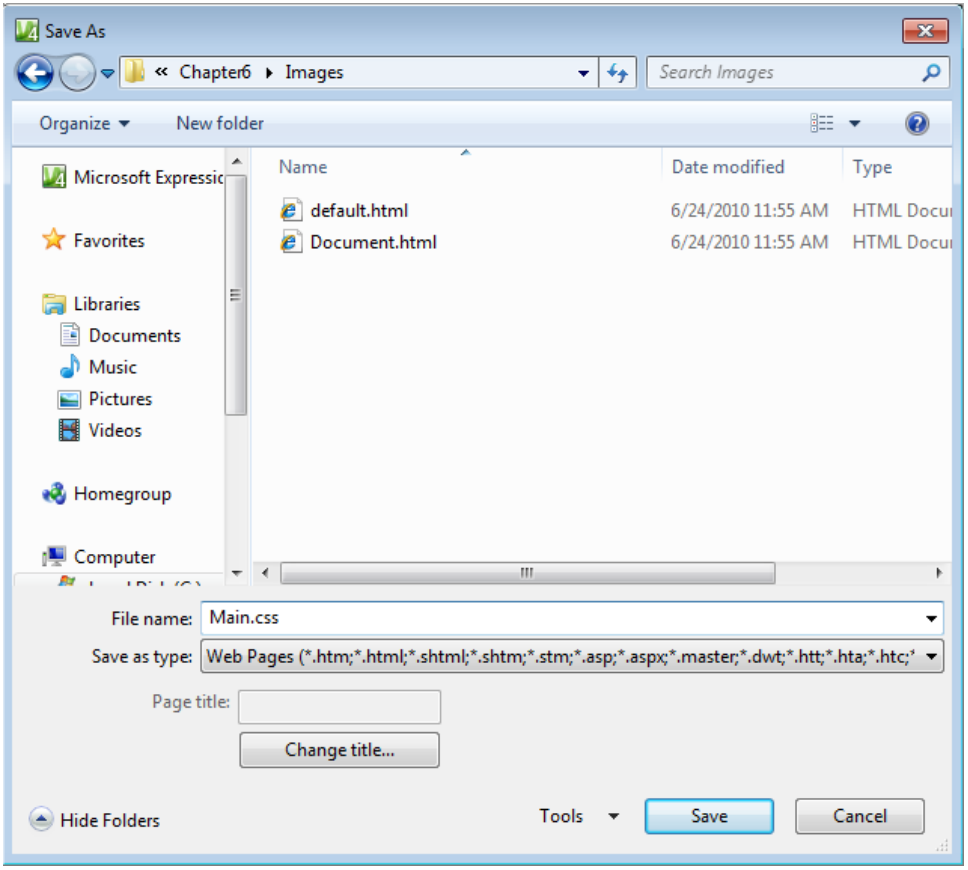
7. In the Folder List pane, press Ctrl while clicking default.html, Document.html, LazyDogText.txt, and template.design to select them all, and then drag them into the Images folder.

   By dragging all the files from the root of the site into the Images folder, you've cleaned up the root folder and you will make the process of building a site within this folder structure much easier. By keeping a clean and orderly folder structure, you will greatly reduce the difficulty of maintaining and modifying a site over the long-term.

Tip Although the folders and files in this demonstration are limited, you won't often find this to be the case in a production site. Click the Site View tab at the top of the editing window to put the site into Folders View, which will make your file and folder tasks much easier.

8. Click the drop-down arrow on the New Document button on the Common toolbar and then click CSS. A new cascading style sheet named Untitled_1.css is created.

9. Click the Save button on the Common toolbar, and in the Save As dialog box, double-click the site's Images folder, and then type **Main.css** in the File Name field.



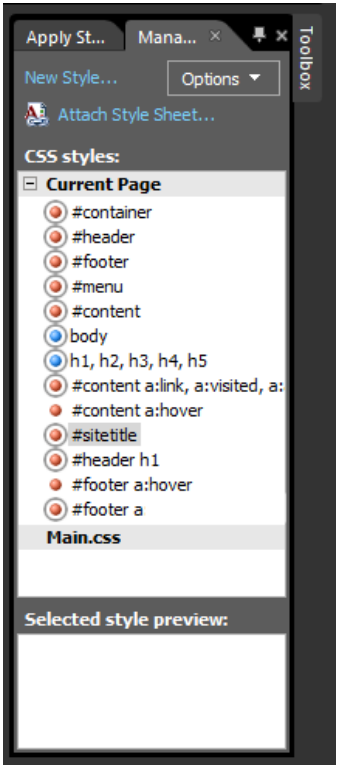
10. Click Save to save the style sheet in the Images folder and close the dialog box.

Tip Saving a style sheet in your site's Images folder makes it much easier to use images in the CSS file. There isn't any folder path that needs to be considered, and keeping the style sheet in the same folder as the images is a good technique to make using images in your CSS easier.

Up to this point, you've been using Expression Web to write the CSS code into the HTML file you've been creating your layout on. This has been the easiest and most efficient way so far, but in the next steps, you will transfer that code to an external style sheet to make the rest of your work easier.

11. Switch to the default.html file that you've been working on. Click Format, select CSS Styles, and then click Attach Style Sheet. In the Attach Style Sheet dialog box, click the browse button. Browse into the site's Images folder, double-click Main.css, and then click OK.

12. To the right of the Apply Styles tab in the Apply Styles panel that you've been using, you will see the Manage Styles tab. Click the Manage Styles tab to activate that panel. Click the thumbtack icon on the Toolbox panel to autohide it, because it's not necessary at this time and it'll give you more space for the Manage Styles panel.

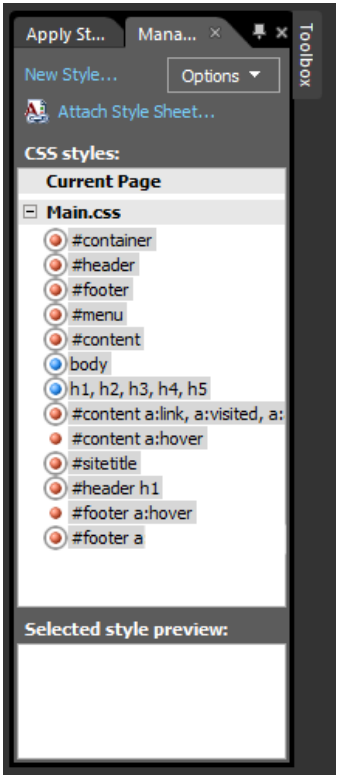


You will use this tab to easily drag the styles from the default.html file into the newly created and linked external CSS file.

13. Click the first style entry beneath the Current Page heading in the Manage Styles panel and then press Shift while clicking the last entry to select all the entries at once.

For this screen shot image, the *#container* is *#footer a:link, a:visited, a:active*.

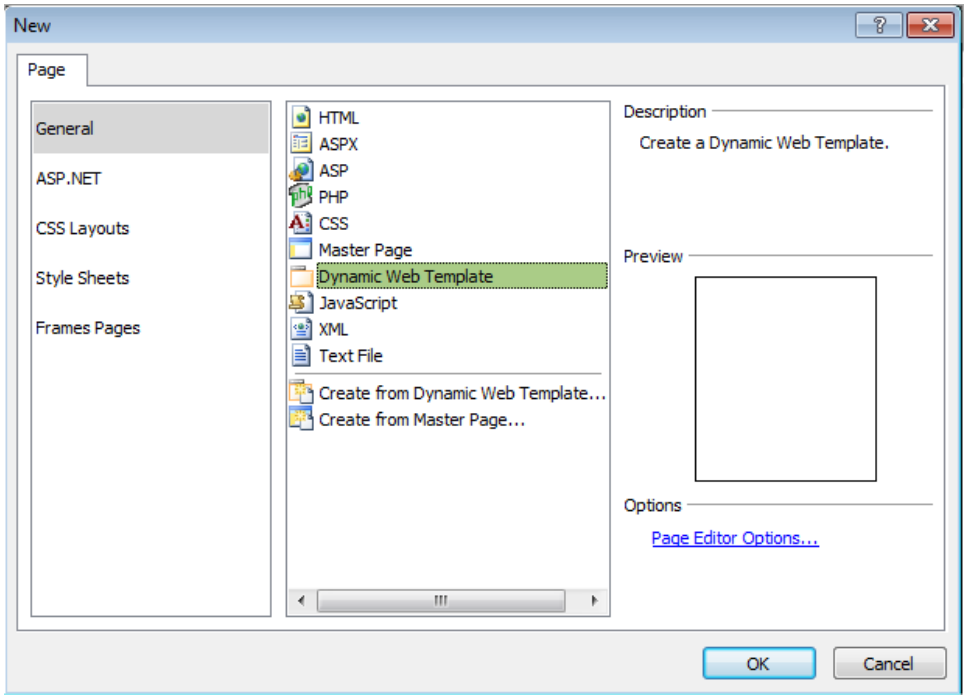
14. Drag the selected style entries onto the Main.css heading below them and release them there.



All the styles you created are moved from the original location within the page to the external style sheet.

15. From the File menu, select Save All.

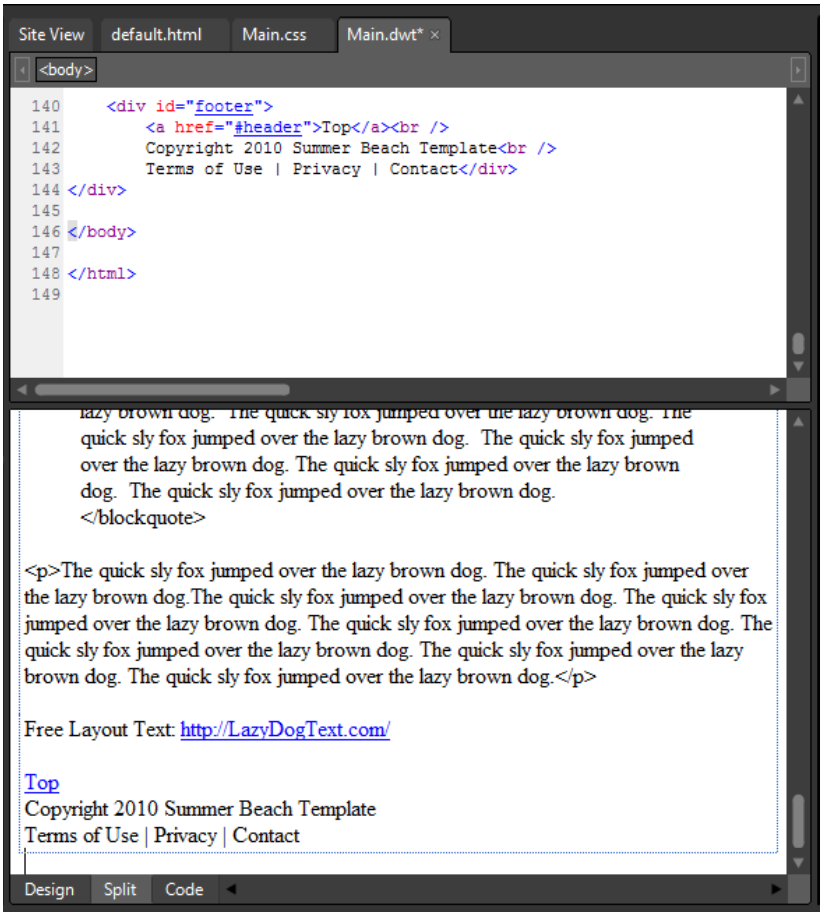
16. Click the drop-down arrow on the New Document button and then click Page. In the New dialog box, in the General category, click Dynamic Web Template.



17. Click OK. A new Dynamic Web Template named Untitled_1.dwt is created. Click the Save button on the Common toolbar, and in the Save As dialog box, navigate to the site's Images folder, type **Main.dwt** in the File Name field, and then click Save.

18. Click the default.html tab at the top of the editing window and set your cursor in the Design pane of the page. Click the drop-down arrow on the <*body*> tag button on the Quick Tag Selector, and then click Select Tag Contents. Right-click the selected code in the Code pane, and in the context menu, choose Copy.

19. Click the Main.dwt tab at the top of the editing window to make that page the active document. Set your cursor in the Design pane of the page, click the drop-down arrow on the <*body*> button on the Quick Tag Selector, and then click Select Tag Contents. Right-click the selected code in the Code pane, and in the context menu, choose Paste.

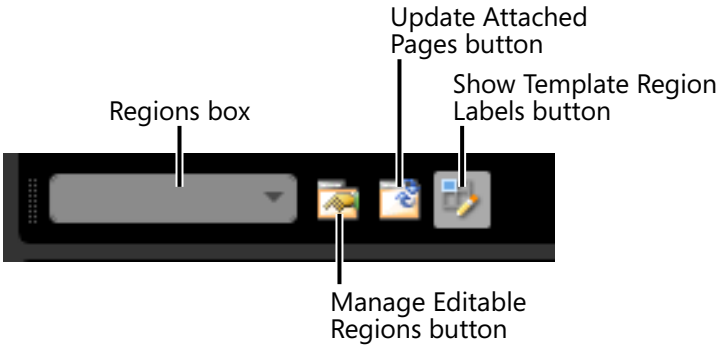


All the divisions you worked on in the default.html page have now been transferred in one easy move to your new DWT. It won't look like the template you have worked on though, because the style sheet isn't attached yet. You'll take care of that in the next step.

20. Click Format, select CSS Styles, and then click Attach Style Sheet. Click Browse in the Attach Style Sheet dialog box, browse into the site's Images folder, double-click Main.css, and then click OK in the Attach Style Sheet dialog box.

As soon as the Main.css style sheet is attached, your Dynamic Web Template will look identical to the HTML file you've been working with.

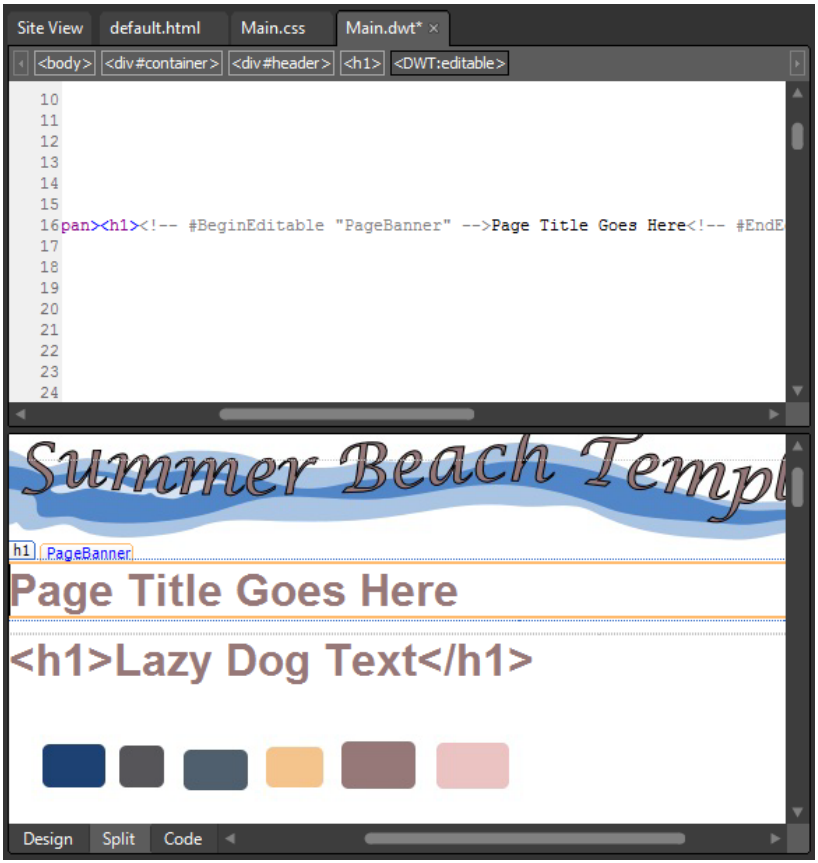
21. Click Save on the Common toolbar. Click View, select Toolbars, and then click Dynamic Web Template. The DWT toolbar appears at the top of your workspace.



The Dynamic Web Template toolbar is fairly small, but it makes working with setting up editable regions easier because every major operation is represented by a button, as opposed to selections several levels into the Format menu. In the next few steps, you will set up two Editable Regions in the Dynamic Web Template.

22. Set your cursor in the Design pane of the first *h1* that reads Page Title Goes Here. On the Quick Tag Selector, click the drop-down arrow on the <*h1*> tag, and then click Select Tag Contents.

23. Click the Manage Editable Regions button on the Dynamic Web Template toolbar. In the Editable Regions dialog box, type **PageBanner** in the Region Name field, click Add, and then click Close.



Notice what has occurred in the page. Instead of putting the *<h1>* tag and its contents inside the Editable Region, you have the *<h1>* tags outside of the Editable Region, and the region is inside the tag pair. This will prevent a user from using a tag other than the *<h1>* in this semantically critical page area.

24. Set your cursor in the content area of the page. Click the drop-down arrow on the *<div#content>* button on the Quick Tag Selector, and then click Select Tag Contents.

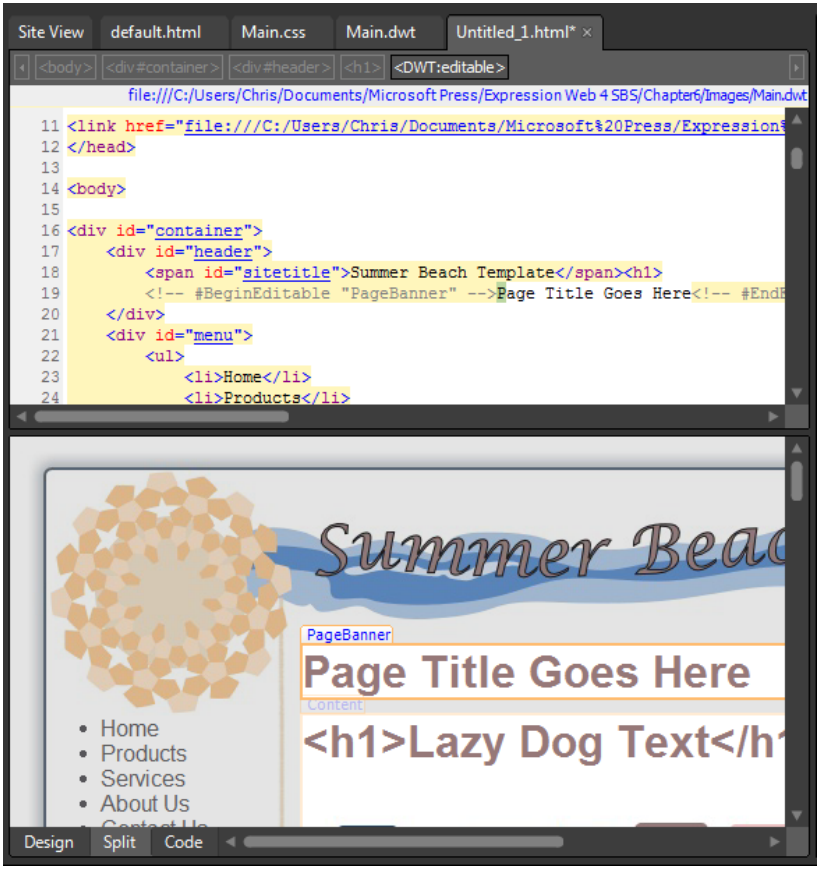
25. Click the Manage Editable Regions button on the Dynamic Web Template toolbar. In the Manage Editable Regions dialog box, type **Content** in the Region Name field, click Add, and then click Close.

    Similarly to how you added the Editable Region inside the *<h1>* tag in the previous step, this editable region is inside the content *div* tag. By setting up the Editable Region this way, it will prevent a user from inadvertently deleting or changing the content division's tag in any way. Although you may be the only designer to ever work on a site you build, using practices such as this will go a long way in keeping the site uniform and durable in any editing situation.

26. Right-click the Code pane, and from the context menu, choose Select All. Right-click the Code pane again and choose Reformat HTML from the context menu. Click Save on the Common toolbar.

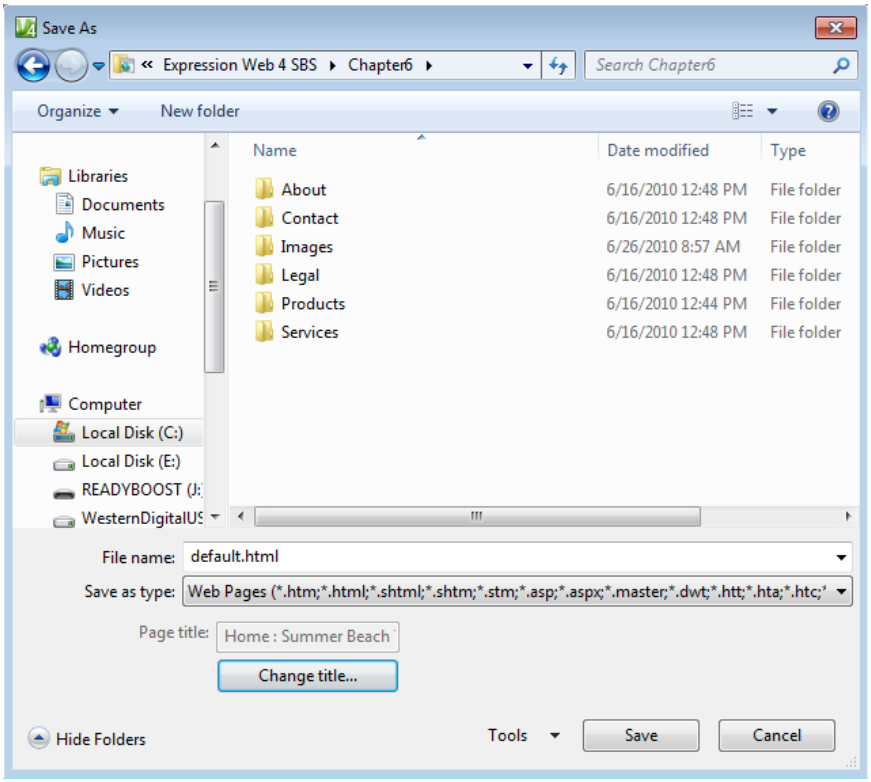
Tip You are about to begin adding pages to the site based on this DWT. If you want to add custom Meta tags to the <head> section's Editable Region, you will achieve easier and more consistent results if you do so now, as opposed to after the pages are created based on the Dynamic Web Template.

27. From the File menu, point to New, and then click Create From Dynamic Web Template. In the Attach Dynamic Web Template dialog box, double-click Main.dwt in the site's Images folder, and then click Close on the file update alert.



A new file based on the Dynamic Web Template, named Untitled_1.html is created in your workspace.

28. Click the Save button on the Common toolbar. In the Save As dialog box, navigate up one level to the root of your site. Make sure that the File Name field reads default.html and then click the Change Title button. In the Set Page Title dialog box, change Summer Beach Template to Home: Summer Beach Template, and then click OK.



Obviously, if you were building a production site, Summer Beach Template would be the name of the site or text relative to the particular site you were building.

29. Click Save in the Save As dialog box to save your new site's new home page as default.html in the root folder.

30. Right-click default.html in the root of the Folder List panel and then, in the Context menu, choose Copy.

31. You now have a copy of your default.html page on your Clipboard. Next, you will make it the default file for each of the folders in the root of your site.

32. Right-click the About folder in the Folder List panel and then, from the context menu, select Paste. Repeat this process for each of the remaining folders: Contact, Legal, Products, and Services. The Images folder already has a default file in it, so you can skip that one.

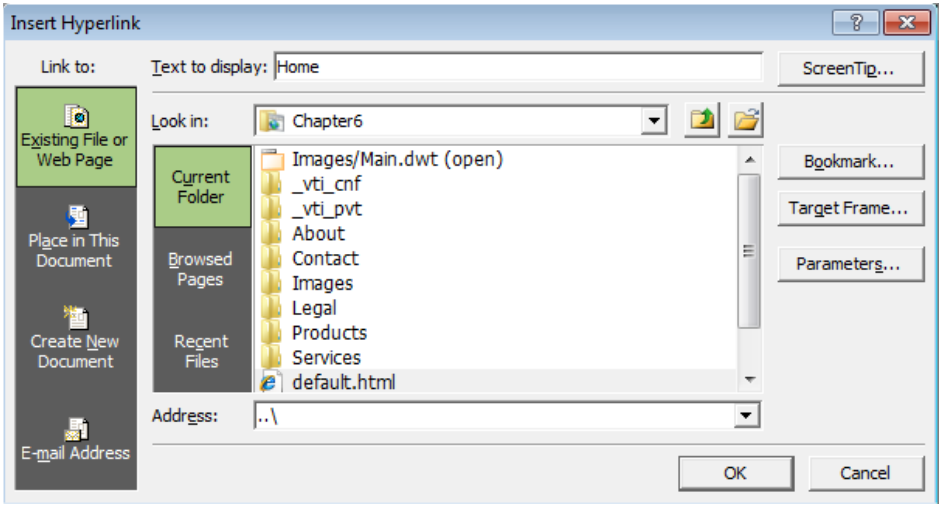
Tip Click a folder in the Folder List panel and then press Ctrl+V on your keyboard to quickly paste a file into it. It's much faster than using the context menu.

You now have a default file in each of your site's folders and you're just a few steps away from having a working navigable site.

33. Using the page tabs above the Quick Tag Selector, close all the open pages except for the Main.dwt file.

Tip You've closed all the open pages except for the DWT because you're about to set up the navigation for the site. These updates automatically occur in closed pages, but open pages would need to be saved manually.

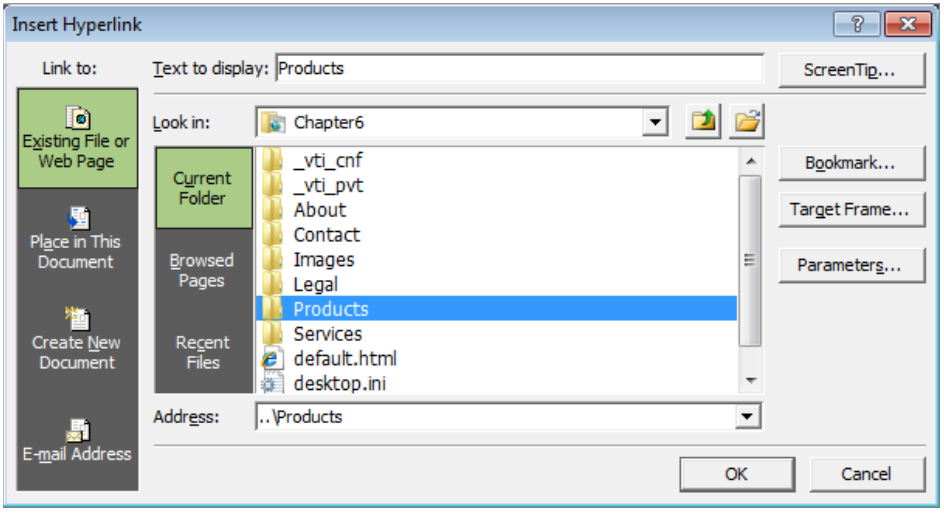
34. In the Menu division of your page, double-click the list item text "Home" to select it. Right-click the selected text, and choose Hyperlink from the context menu. In the Insert Hyperlink dialog box, click the default.html file in the site's root. The Address field will be populated with "..*default.html*." Backspace through the page name, erasing characters until the Address field contains only the two periods and the backslash (**..**).



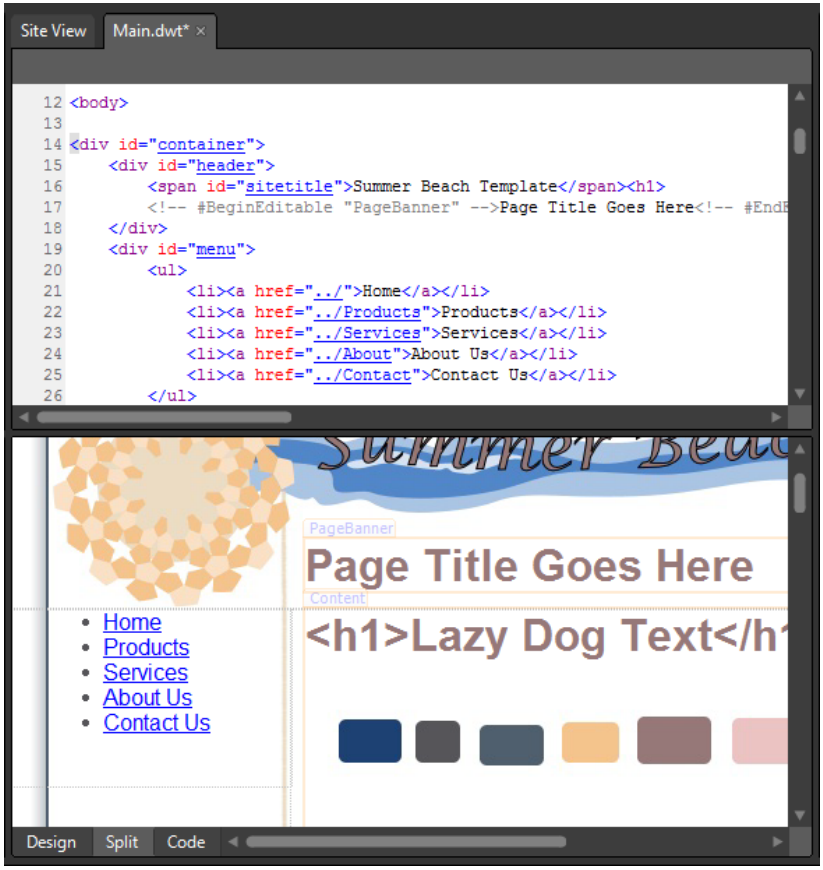
For the navigation in this site, you won't use any page names or extensions. Instead, you will use folder path navigation. It results in cleaner addresses in visitor's address bars, and if you decide to change the default document, you can easily switch that in a folder without having to change the navigation at all. For instance, if you decide to use a dynamic page such as a PHP or ASP.NET page, you can easily put a default.php or default.aspx file in a folder, delete or rename the default.html file, and therefore upgrade the page without changing the site navigation. Search engine placement and visitor's bookmarks stand a much better chance of remaining correct using this method.

35. Click OK on the Insert Hyperlink dialog box to set the link.

36. Double-click Products to select the list item text. Right-click the selected text and then select Hyperlink in the context menu. Click the site's Products folder to select it. The Address field will be populated with "..\Products," rather than "..\Products\default.html."



37. Continue hyperlinking the remaining list items (Services, About Us, and Contact Us) to their respective folders.



Although the navigation at this point clearly needs some adjustment for its visual appearance, take a look in the Code pane representation. It's clean and simple, and what is a navigation bar really other than a list of links? This navigation will be easy to understand for search engines, screen readers, and conventional visitors alike.

38. Click New Style in the Manage Styles panel. In the New Style dialog box, type **#menu ul** into the Selector field. Set the Define In field to **Existing Style Sheet**, click the Browse button beside the URL field, and in the Select Style Sheet dialog box, double-click the Main.css file in the site's Images folder.

39. Click the Box category and set both the Padding and Margin fields to **0px**, leaving the Same For All check box selected.

40. Click the List category and set the List-Style-Type field to **none**. Click OK to set the style and close the dialog box.

You've set these specifications to remove the "bullets" from the list items in your menu and remove any inherent padding and margin that the list could be given by browsers.

41. Click New Style on the Manage Styles panel. In the New Style dialog box, type **#menu li a** into the Selector field. Make sure that the "Define In" field is set to **Existing Style Sheet** and the URL field shows **Main.css**.
42. Set the Font-Weight field to **bolder**, click the drop-down beside the Color field, and then click More Colors. In the More Colors dialog box, click the Select button, and then click the pink color rectangle in the Design view of the colors.jpg image. Beneath the Text-Decoration field, select the check box beside none, to remove the link's inherent underline.
43. Click the Block category and set the Text-Align field to **center**.
44. Click the Background category and set the Background-Color field to **#FFFFFF** (white).
45. Click the Border category. Leaving the Same For All check boxes selected, set the Border-Style to **solid**, the Border-Width to **2px,** and the Border-Color to **#EBC3C2** (the same as the font color).
46. Click the Box category. Leaving the Same For All check boxes selected, enter **2px** for the padding and **10px** for the margin.
47. Click the Position category and enter **140px** in the Width field.
48. Click the Layout category and set the Display field to **block**.

    You're setting the Display field to block because it will make the links act more like buttons where the user doesn't need to click the text, but can click anywhere on the list item.
49. Click OK to close the dialog box and set your style changes.

Your links look a lot more like buttons now. There's just the matter of assigning a different color to them for their hover state so that the user will get a little visual feedback when they mouse-over them.

50. Click New Style on the Manage Styles panel and type **#menu li a:hover** into the Selector field. Make sure the Define In field is set to **Existing Style Sheet** and the URL field shows **Main.css**.

51. Click the drop-down arrow beside the Color field in the Font category and then click More Colors. Click the Select button in the More Colors dialog box, and then click the mauve color rectangle in the Design view rendering of colors.jpg.

52. Click the Border category and, leaving the Same For All check boxes selected, enter **solid** in the Border-Style field, **2px** in the Border-Width field, and **#967878** (the same color as the font) in the Border-Color field. Click OK to set the styles and close the dialog box.

53. Scroll the Design pane down to the footer of the page and select the words "Terms of Use." Right-click the selected words and then, from the context menu, choose Hyperlink. Click the Legal folder and then click OK.

54. Select the word Privacy and right-click the selection. Select Hyperlink from the context menu, click the Legal folder, and then click OK.

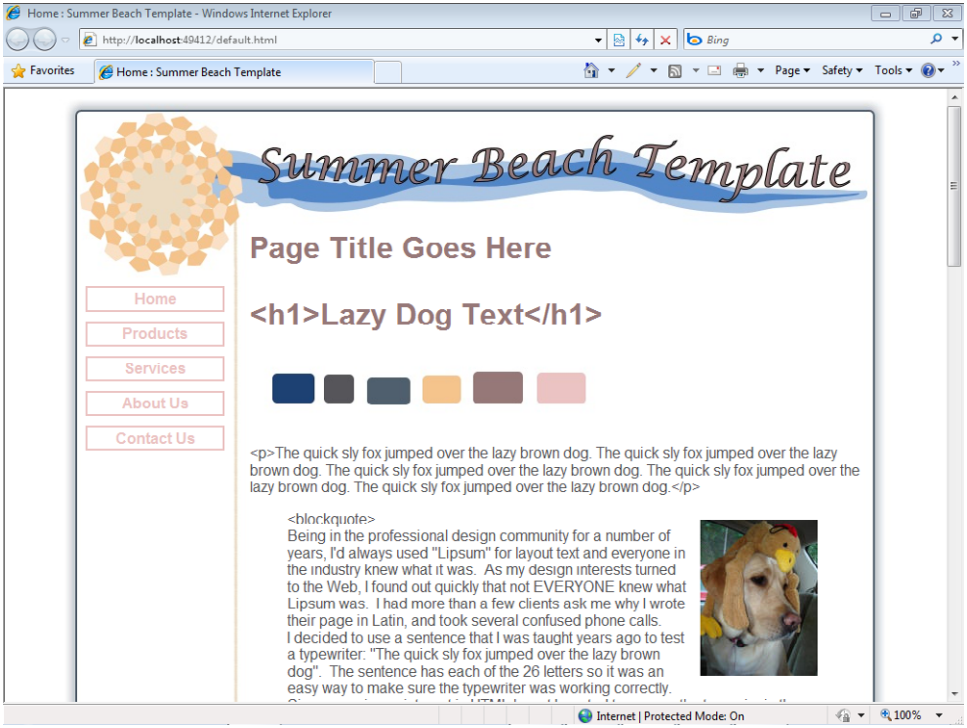
55. Select the word "Contact" and right-click. From the context menu, click Hyperlink, click the Contact folder, and then click OK.

56. Click Save on the Common toolbar. In the Save Embedded Files dialog box, click OK to let the Main.css file overwrite and save. Click Yes on the file update alert to allow Expression Web to update all the files attached to the Dynamic Web Template you're working on, and then click Close on the confirmation alert.

57. Choose Site Settings from the Site menu, and then click the Preview tab in the Site Settings dialog box. Select the For All Web Pages option beneath the Use Microsoft Expression Development Server label, and then click OK to set your change and close the dialog box.

    Because this site is using folder paths for the navigation, you will need the Expression Development Server to preview the pages. If the Development Server wasn't used, the default behavior would be for your folder-based links to open in Windows Explorer and show you the folder contents. It takes a server to know what the default document of a folder is.

58. Click default.html in the Folder List panel and then click the Preview button on the Common toolbar to open the page in a browser.



59. Try all the links in your navigation structure, both in the menu division and footer division of the pages, to make sure they behave as expected, then close the open browser and return to Expression Web.

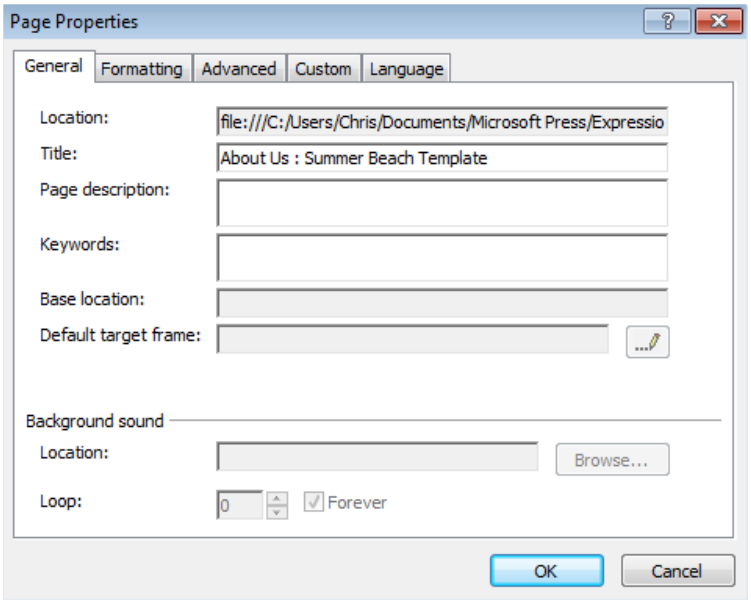
As you were clicking around in the browser, you probably noticed that all the pages were virtually identical. The identical nature of them may even have made you think the pages weren't changing, but keep an eye on the address bar and you will see that they are, in fact, changing. Although this section isn't about writing content for pages, we will wrap it up by making each of the pages unique by adding their page title and the page banner *<h1>* element.

60. Double-click default.html in the Folder List panel to open it for editing. Change the text "Page Title Goes Here" to **Our Home Page**.

By having the page title, which you added when you first created the page, the first *<h1>* element on the page, and the URL in the address bar all relate to each other, the search engine placement of the page will be much greater than if these steps weren't taken.

61. Expand the About folder in the Folder List panel and then double-click its default.html file. Change the "Page Title Goes Here" text to **About Us**.

62. Right-click the Design pane of the page, and from the context menu, choose Page Properties. On the General tab, change the title from Home: Summer Beach Template to **About Us: Summer Beach Template**.



The Page Properties dialog box also provides a convenient place to enter a page description and keywords. If you fill in these fields, you will see these meta tags written into the <*head*> section of your page. Currently, keywords aren't relied on by search engines, but a page description tag will appear in search engine results pages.

63. Type **This is the About Us page of the Summer Beach Template site**, into the Page description field, and then click OK to set your changes and close the dialog box.

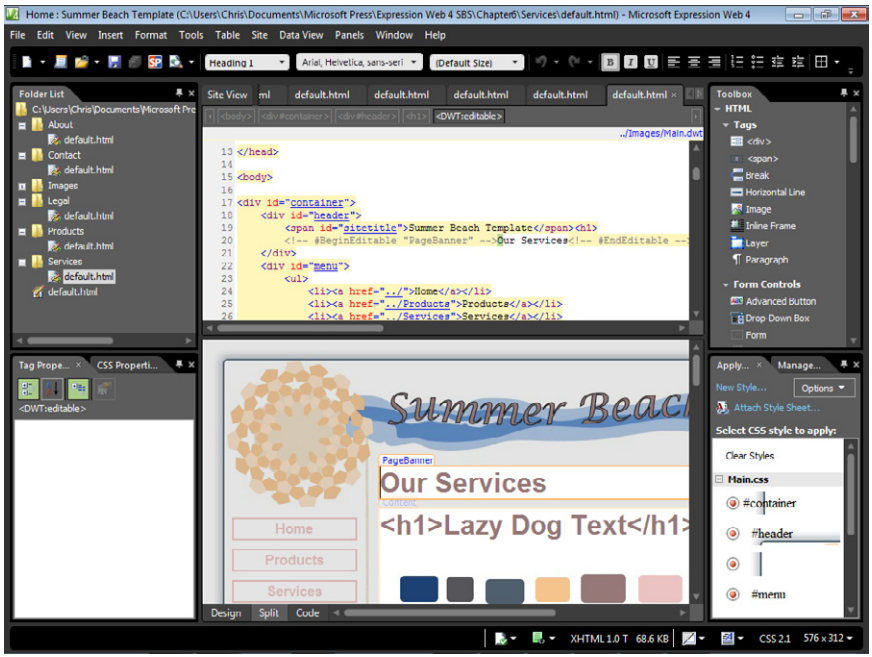
64. Expand the Contact folder in the Folder List panel and then double-click the default.html file to open it for editing. Change the "Page Title Goes Here" text to **Contact Us**. Right-click the Design pane and from the context menu, select Page Properties. Enter a page title of **Contact Us: Summer Beach Template**, insert a Page description in the appropriate field, and then click OK.

65. Continue changing the default <*h1*> text, Title, and Page descriptions in each of the remaining pages: Legal/default.html, Products/default.html, and Services/default.html.

Tip If you open a page and it doesn't match the other pages, it is most likely a rendering issue in Expression Web. Press F5 on your keyboard or select Refresh from the View menu before trying to remedy such an issue by other means.

66. Click File, and then click Save All to save all the page changes you've made while entering heading text, titles, and descriptions.

67. From the Panels menu, click Reset Workspace Layout, to return all your panels to their original state, and then click View, select Toolbars, and then click Dynamic Web Template to close that toolbar.



Although there would still be the significant outstanding work of creating and tuning content for the pages of this site, what you have at this point is a multiple page, custom designed, and fully navigational site. The techniques that you've learned and employed in this section can serve as the basis of how you lay out the folder architecture and navigation of sites you build in the future. By using the Dynamic Web Template and cascading style sheets, this site will be easy to expand, manage, and maintain.

Note If you're not continuing directly to the next exercise, you can close all the open pages (select Close All Pages from the Window menu) and then exit Expression Web.

Styling for Alternative Media

One of the really great things about cascading style sheets is the separation of content and presentation. More than just a "buzz phrase," this section will show you exactly what that means in a tangible way. You will create a style sheet specifically for print and attach it to your master page using the media attribute in the style sheet link. This is interesting because you don't have to change a single line of HTML on your Dynamic Web Template or any of

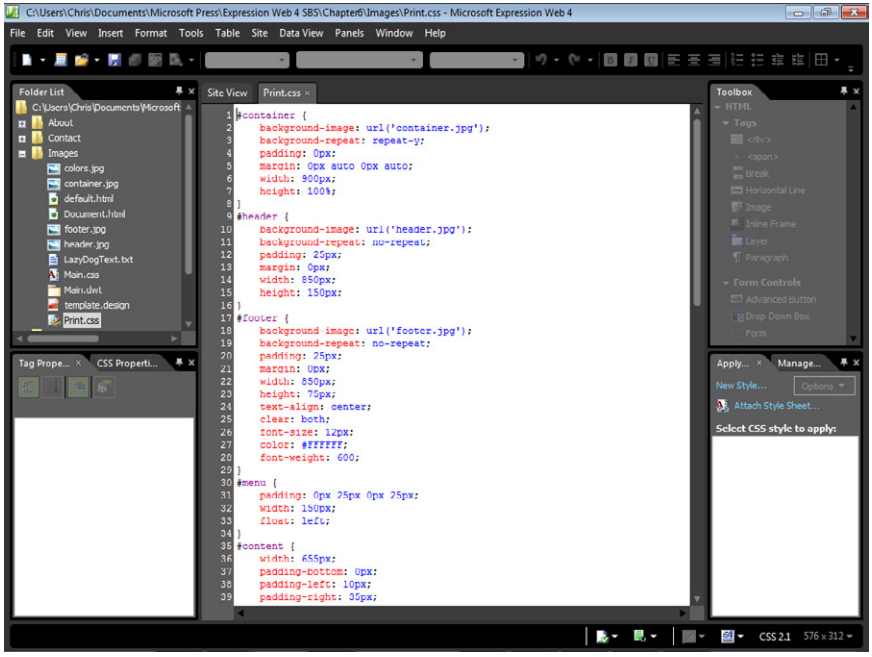
the pages that it's attached to. You also don't need to make additional pages specifically for printing, nor do you need to do any kind of server-side programming. All you will have to do is link the print style sheet to the DWT.

By employing this method, you will be able to make your Web pages print completely differently than they appear on screen. Although it's a simple technique, it's an underused feature. Many sites that could benefit from a clean print layout don't even make an attempt. So if you employ this technique for your own and/or your client's sites, you will be providing the visitors with a benefit that most sites simply overlook.

Create a cascading style sheet for the print version of your pages

Note Using the Chapter6 site that you created earlier, open the Main.dwt file in the Images folder if it isn't still open from the previous exercise.

1. In the Folder List panel, expand the Images folder and then right-click Main.css. Choose Copy from the context menu, then right-click again, and select Paste from the context menu. A copy of this file named "Main_copy(1).css" is pasted into the Images folder.
2. Right-click the newly created copy and choose Rename from the context menu. Rename the file to **Print.css**, and then double-click the file to open it in the editing window.



The next few steps will give you an opportunity to edit a style sheet directly. Until this point, you've relied on the Style dialog box in Expression Web. Although you could use the Style dialog box here, it will be much more efficient to just directly edit this style sheet. You'll be removing much more than you'll be adding so it's not going to take very long.

3. Starting with the *#container* ID, remove all specifications except for *height: 100%*, and change the width from 900px to **100%**.
4. Remove all specifications from the *#header* ID and enter **width: 100%.**

Tip Notice that as you begin typing in a style block, options begin to pop up over your workspace. This is Intellisense for CSS. As you learn how to manually write CSS, this tool will become very helpful.

5. Edit the *#footer* ID to remove all specifications except:

```
width: 100%;
text-align: center;
clear: both;
font-size: smaller;
color: #000000;
```

6. Remove all specifications from the *#menu* ID and give it the style **display:none;.**

   The reason for *display:none;* on the menu division is simple: what good is a navigation menu when a Web page is printed?

7. Change the *#content* ID to:

```
width: 100%;
text-align: left;
```

8. Change the body selector to:

```
font-family: "Times New Roman," Times, serif;
font-size: medium;
color: #000000;
```

In print, the serif fonts are more readable than the sans-serif fonts commonly used on screen.

9. Change the h1, h2, h3, h4, h5 selector to **color: #000000;**.

10. Change the #content a:link, a:visited, a:active selector to **#content a** and give it the following styles:

```
color: #000000;
text-decoration: none;
```

Because readers won't be able to click the link, there's really no reason to underline it.

11. Completely remove the *#content a:hover* selector and its specifications, because there's no hover event in printed pages.

12. Edit the *#sitetitle* selector so it has the following styles:

```
width: 100%;
display: block;
font-size: large;
```

This is an interesting point. The original styling of this division contained the site title but not shown on screen because there was a stylized title in the graphics of the site. Because the user won't be printing the background images, we can just style this title to print.

13. Remove the selectors *#header h1* and *#footer a:hover* along with their specifications. They won't be needed so you can just remove them from the style sheet.

14. Edit *#footer a* so it contains only **display: none;**.

There's no need to give footer links to visitors that have printed the page, so that line of CSS will cause them to not even print.

15. Remove the *#menu ul*, *#menu li a*, and *#menu li a:hover* selectors and styles from the style sheet.

16. Although you've already specified that the #menu division is *display:none*, and as such, none of its content should display, there's nothing wrong with keeping a clean, tight style sheet.

17. Right-click the style sheet and select Reformat CSS from the context menu. Then click the Save button on the Common toolbar.

Site View | Print.css ×

```
#container {
    width: 100%;
    height: 100%;
}
#header {
    width: 100%;
}
#footer {
    width: 100%;
    text-align: center;
    clear: both;
    font-size: smaller;
    color: #000000;
}
#menu {
    display: none;
}
#content {
    width: 100%;
    text-align: left;
}
body {
    font-family: "Times New Roman", Times, serif;
    font-size: medium;
    color: #000000;
}
h1, h2, h3, h4, h5 {
    color: #000000;
}
#content a {
    color: #000000;
    text-decoration: none;
}
#sitetitle {
    width: 100%;
    display: block;
    font-size: large;
}
#footer a {
```

You now have a style sheet that specifically hides the page areas that provide no benefit to print, and refined the styles for the page areas that you are keeping to be more print friendly.

18. In the Folder List panel, double-click the Main.dwt file in the site's Images folder to open it in the editing window, and then scroll the Code pane to the *<head>* section of the page.

19. Set your cursor inside this line…

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

    …just after *type="text/css"* and then type **media="screen"**.

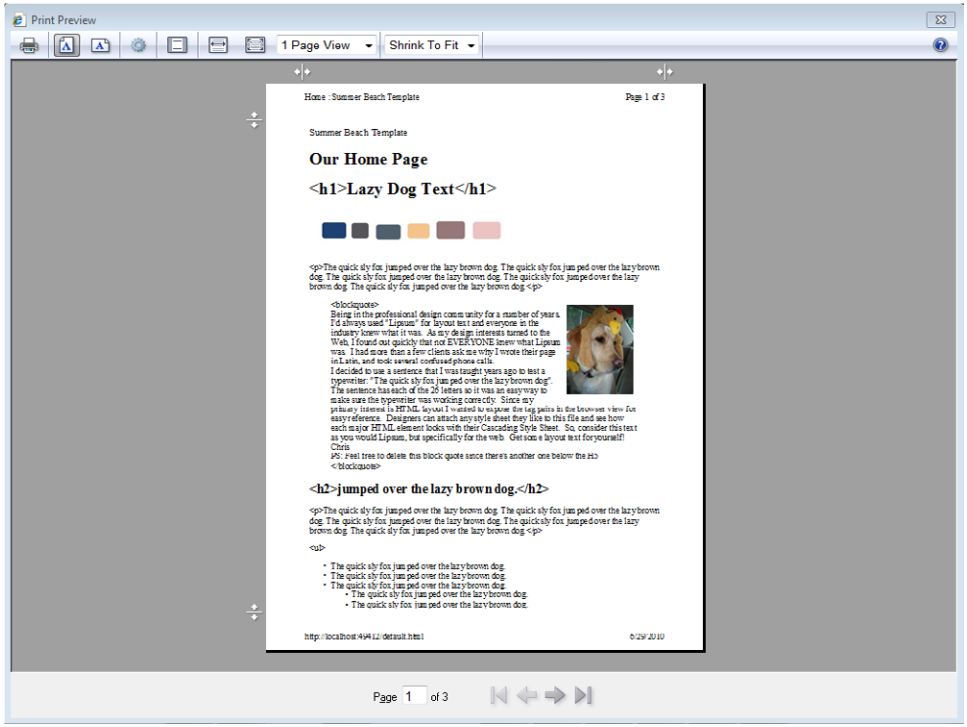
20. Notice that Intellisense pops up as you begin to type this line; when it shows the item you want, just press Enter on your keyboard to enter it automatically.

21. Set your cursor below the line you just edited, and type:
    <link href="Print.css" rel="stylesheet" type="text/css" media="print" />

22. Click Save on the Common toolbar, click Yes on the file update prompt, and then click Close on the update confirmation.

23. Click default.html in the root folder in the Folder List panel, and then click the Preview button on the Common toolbar to open the page in a browser.

24. In Internet Explorer, press the Alt key on your keyboard to show the Menu bar, and then choose Print Preview from the File menu.



Print preview is a real help when you're spending time to get the print version of a page to an exact specification. Besides just saving ink and paper, it's much faster than actually printing the pages.

25. With the Print Preview open, press Escape on your keyboard to return to Internet Explorer, and then close the browser and return to Expression Web.

What you've just done and seen is a good example of some of the benefits that a CSS-based design affords the designer because of the separation of content and presentation. By simply adding a style sheet for print, the designer is able to provide a completely different appearance to the page than would be present on the screen. It's a powerful technique that requires very little effort and shouldn't be overlooked. Although you used the media types Screen and Print, there are 10 different media types you can use:

- all
- aural
- braille
- embossed
- handheld
- print
- projection
- screen
- tty
- tv

Note Close any open pages, and then select Exit from the File menu to close Expression Web.

Key Points

- As the basis of every site, the graphics are key to creating an attractive and usable template.
- Expression Web provides tools to help you lay out your graphics into an HTML template.
- By providing thoughtful site architecture, the designer can improve search engine results, while at the same time providing better accessibility to visitors.
- Using CSS, the designer can provide different appearances for pages on the screen, in print, and a number of other media types.

Chapter 8

Adding Functionality with jQuery and PHP

After completing this chapter, you will be able to:

- Use jQuery in Expression Web
- Use the Expression Web PHP tools

Microsoft Expression Web 4 provides support not only for Microsoft technologies, but also for other popular Web development tools. This chapter shows you how to take advantage of the jQuery JavaScript library and the PHP Web development language using Expression Web 4.

Important Before you can use the practice files in this chapter, you need to download and install them from the book's companion content Web site to their default location. For more information about downloading and installing the practice files, see the "Code Samples" section at the beginning of this book.

Troubleshooting Graphics and operating system-related instructions in this book reflect the Windows 7 user interface. If your computer is running Windows XP or Windows Vista and you experience trouble following the instructions as written, please refer to the "Information for Readers Running Windows XP or Vista" section at the beginning of this book.

Using jQuery in Expression Web

jQuery is a lightweight JavaScript library that makes authoring JavaScript faster and easier than ever. It's currently very popular among designers and developers alike, because it's easy to use, flexible, and has a plug-in ecosystem that lets you add additional functionality.

With jQuery, you can write simplified JavaScript that's compatible with multiple browsers, letting you create functionality more easily than ever before.

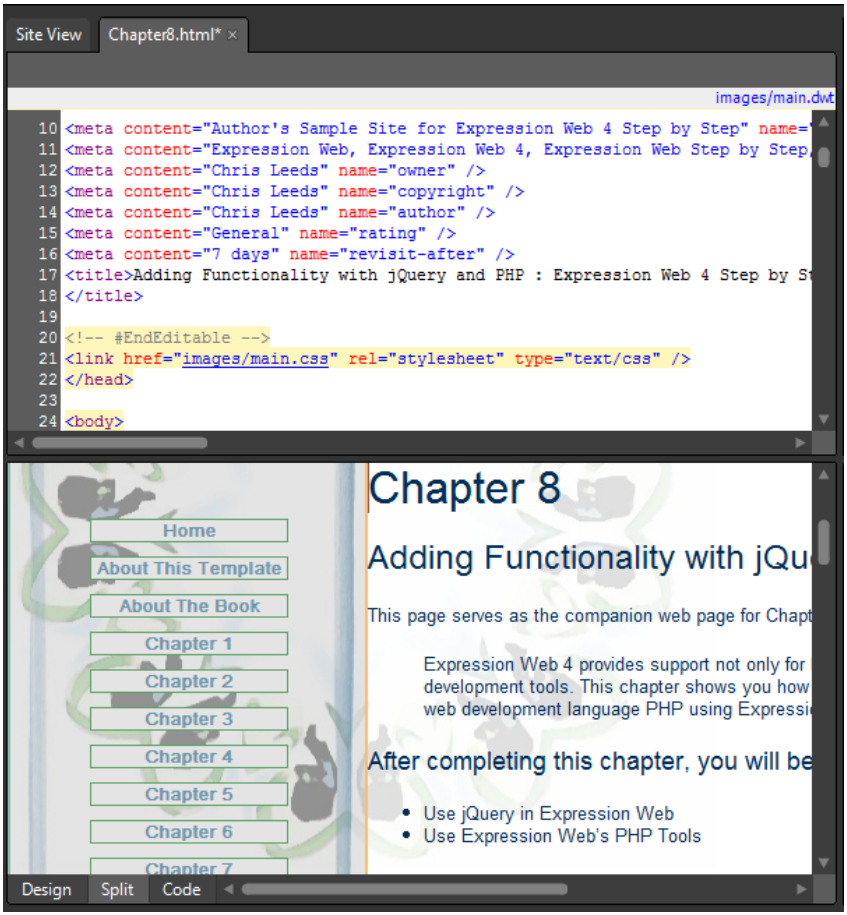
Because jQuery is a framework library it's possible for developers to create "plug-ins" for the framework, which add specialized functionality that extends jQuery for specific needs. Many plug-ins have already been written; you can find a list of them at *http://plugins.jquery.com/*. After installing a plug-in, you can use its features on your pages with little to no coding.

In this exercise, you will write some jQuery-compatible JavaScript and use a jQuery plug-in.

Link a page to a jQuery library, write a simple function, and use a jQuery plug-in

Note Start Expression Web 4 before beginning this exercise. Open the SampleSite site by selecting Open Site from the Site menu, browsing to the companion media's default installation page, and double-clicking the SampleSite folder. With the sample site opened in Expression Web, double-click the Chapter8.html page in the Folder List panel.

1. Scroll the Code pane of Chapter8.html to the *<head>* section and set your cursor just before the closing *<!-- #EndEditable -->* in the "doctitle" editable region and then press Enter on your keyboard to break to a new line.



The first thing you need to do when using jQuery is link to the library itself; that's what you will do in the next few steps.

2. With your cursor on a new line in the editable region within the page's <*head*> section, begin to type **<script**... and the Expression Web Intellisense will appear. Each time the Intellisense pop-up displays the code fragment you want, press Enter on your keyboard. When complete, you will have a line that looks like this:

```
<script type="text/javascript" language="javascript" src="files/jquery-1.4.2.min.
js"></script>
```

In this example, you are linking to the jQuery file that resides physically in the SampleSite's folder structure—specifically, the site's /files folder. Using a local copy of the jQuery library as in this example is just one option for linking to the library; many designers link to the jQuery library via an external URL. Both Google and Microsoft host this library, and anyone can link to it freely. The following example script tags would link a page to the Google or Microsoft hosted jQuery library:

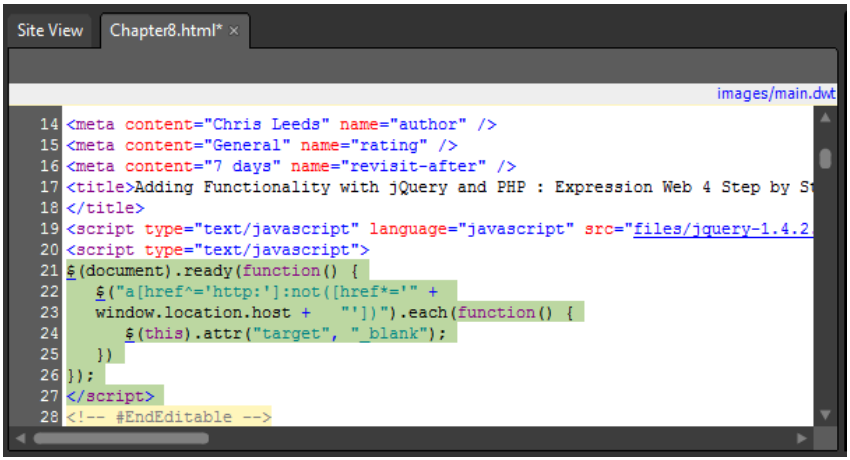
```
<script language="javascript" type="text/javascript" src=
 "http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js">
</script>

<script language="javascript" type="text/javascript" src=
 "http://ajax.Microsoft.com/ajax/jQuery/jquery-1.3.2.min.js">
</script>
```

Note In this example, you are linking to the jQuery library locally so that you don't need an active Internet connection to complete the examples.

Tip If you're going to use the jQuery library in many or all of your pages, you can add the link to a template file so it will be available automatically on all the pages based on that Dynamic Web Template (DWT) or ASP.NET master page.

3. The next step in this jQuery example is to write actual script into the page. Press Enter on your keyboard to create a new line beneath the script tag that you inserted. Enter the following code:

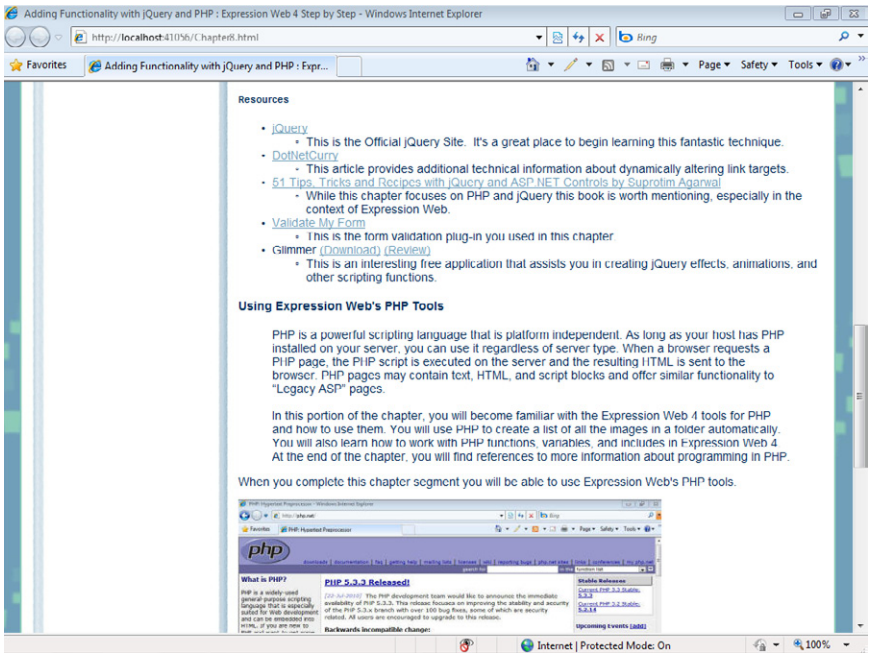
```
<script type="text/javascript">
$(document).ready(function() {
   $("a[href^='http:']:not([href*='" +
   window.location.host +   "'])").each(function() {
      $(this).attr("target", "_blank");
   })
});
</script>
```



If you don't want to type that entire block of code, you can continue this exercise by copying and pasting from within the comment tags in the file /files/external_blank.js, or simply link to the file in the same way that you did the jQuery library:

```
<script language="javascript" type="text/javascript"
  src=" files/external_blank.js"></script>
```

4. Click Save, and then click Preview on the Common toolbar to open your page in a browser. Scroll down to the jQuery Resources list.



5. Click a few of the hyperlinks in the list. Notice that they open in a new window. Return to the browser view of Chapter8.html and click one of the page links on the upper-left navigation area of the page. Notice that the internal links now open within the *current* browser window. The JavaScript you added works with the jQuery library to read all the hyperlinks in the page, and if they contain an HREF attribute that begins with "http," gives them a *_blank* target attribute. The result: All the external links open in a new window, and all the internal links open conventionally.

Tip Using jQuery or another method to alter HTML elements is particularly useful in an instance where the actual content you want to alter doesn't exist in your source code or isn't under your control. In the previous chapter, the Data View isn't compiled in Expression Web. The content may change constantly. The Data View is designed to consistently render content the same way even with a potentially changing XML file, so altering the links dynamically is virtually required. Try steps 1 through 3 on Chapter7.html and take note of how you were able to change the behavior of the hyperlinks in the Data View when you test it in the browser.

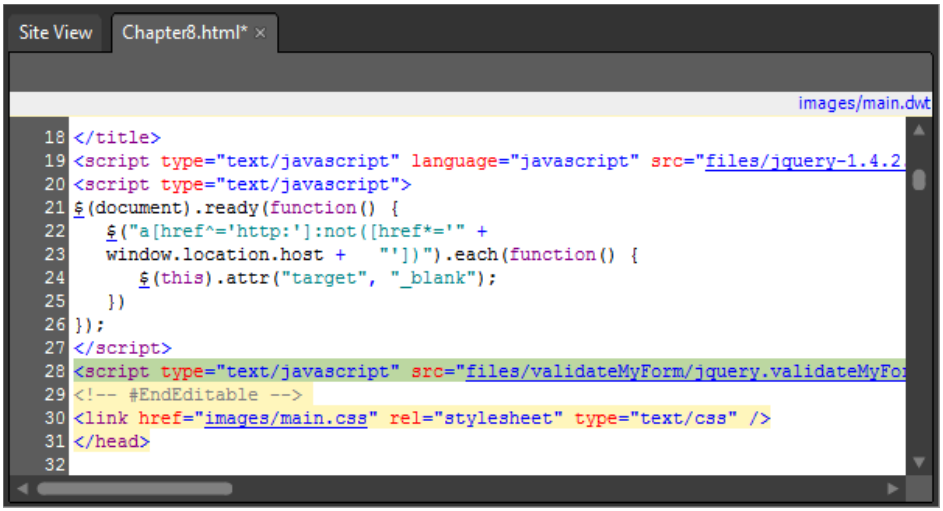
6. Close the browser window and return to Expression Web. Click one of the jQuery Resource links in the Design pane and notice the Code pane. There is no `target="_blank"` attribute in any of the links in the list, which is how you would normally make hyperlinks open in a new window.

In the next exercise, you will link to a jQuery plug-in, which is a pre-written set of functions designed and tested to work with the jQuery Library. You'll use this particular plug-in to validate the input fields of the form you created in Chapter 7, "Adding Client-Side Functionality."

Link to a jQuery plug-in and use it to validate user input

1. Set your cursor in the <*head*> section of the page where you've been working and press Enter on your keyboard to insert a new line after your most recent script block.
2. Using the Expression Web Intellisense pop-up as you did previously, enter the following script tag, which links to the jQuery plug-in:

```
<script type="text/javascript" src=" files/validateMyForm/jquery.
validateMyForm.1.1.js"></script>
```



To use this plug-in, you must link to it just as you linked to the jQuery library. This plug-in is called Validate My Form. You can find a link to it beneath the Resources heading in the Chapter8.html file.

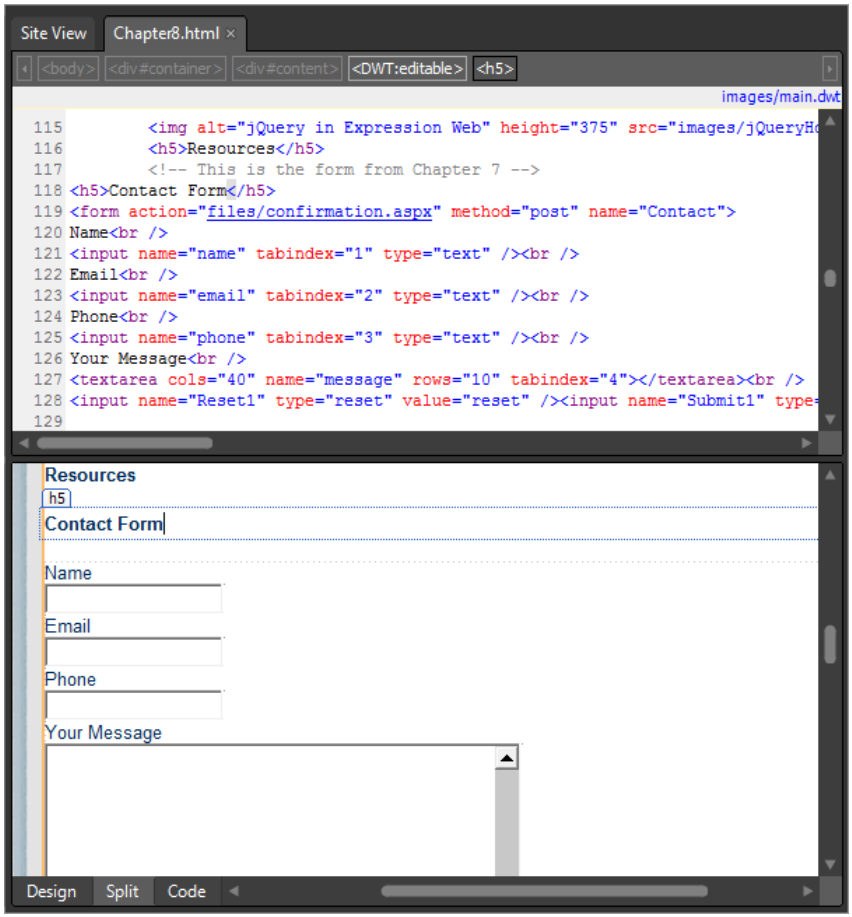
3. In the Code pane, press Enter on your keyboard to start a new line beneath your link to the form validation plug-in and then enter the following script tag and style sheet link:

```
<script type="text/javascript">
<!--
    $(document).ready(function(){
       $("#form1").validateMyForm();
    });
-->
</script>
<link href="files/validateMyForm/css/plugin.css"
   rel="stylesheet" type="text/css" />
```

4. Scroll the Design pane of your page to the Resources heading in the jQuery segment. Set your cursor in the Design pane at the end of the heading text and press Enter on your keyboard to create a new paragraph.

5. In the Folder List panel, expand the site's /files folder and then double-click Chapter7 Form.txt to open it in your workspace. Set your cursor in the code of that file and press Ctrl+A to select all of the text, and then Ctrl+C to copy it all to your Clipboard. Close Chapter7Form.txt and return to Chapter8.html.

6. Set your cursor inside of the new paragraph tag in the Design pane and then click the *<p>* tab on the Quick Tag Selector to select the entire tag. Right-click the highlighted code in the Code pane, and in the Context menu, click Paste. Then click Save on the Common toolbar.

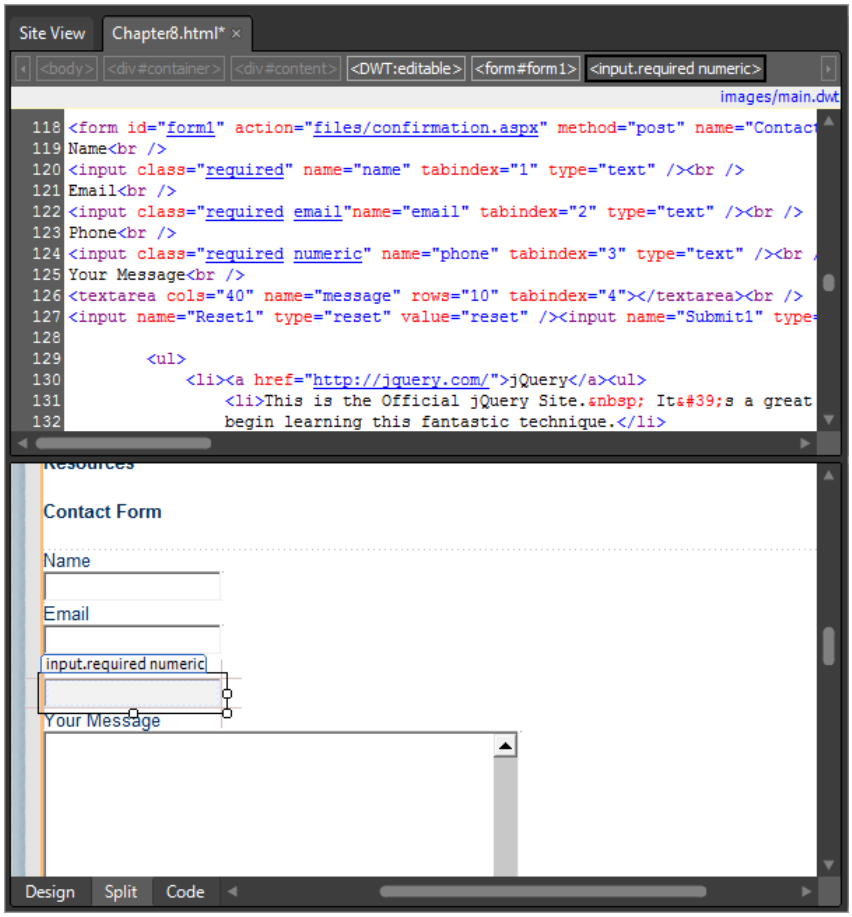


The HTML form from Chapter 7 is now pasted into the page where your paragraph tag was, and the page is saved.

7. Set your cursor inside the Form tag in the Code pane and add the ID attribute **id="form1"** to the tag.

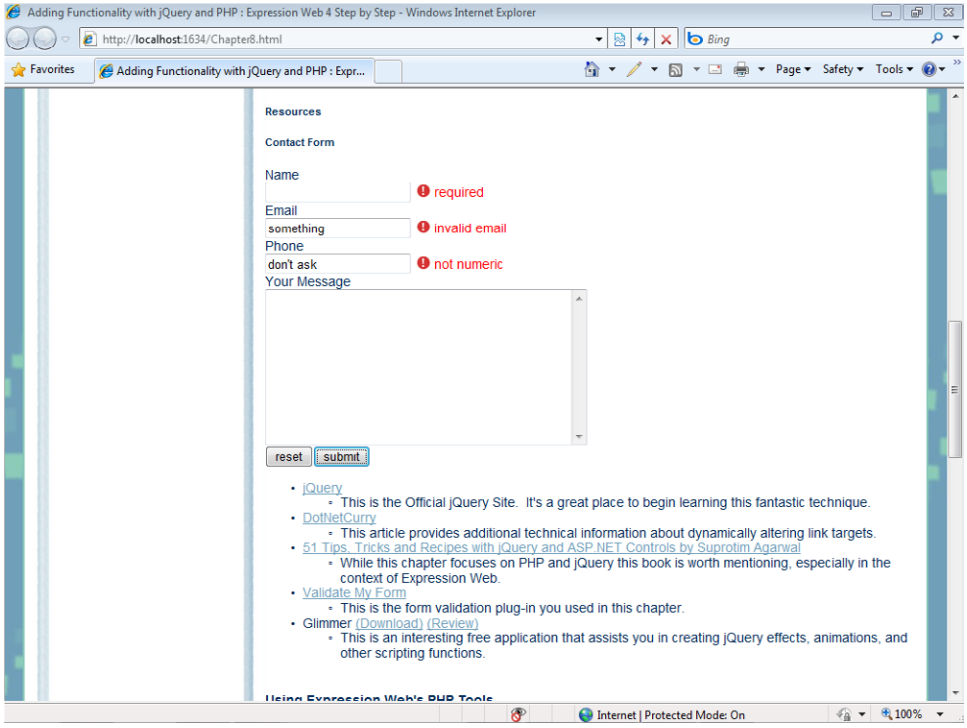
   By adding the ID to the form tag, you have given the plug-in the ability to manipulate the form and apply the validation functionality. This plug-in not only identifies the form by its ID, but enables validation simply by applying a class to the form field in question.

8. Click the Name form field in the Design pane to select it, and then in the Code pane, add **class="required"** to the tag.
9. Click the Email form field in the Design pane and then add **class="required email"** to the tag in the Code pane.
10. Click the Phone form field in the Design pane and add **class="required numeric"** to the tag in the Code pane.



Tip If you find that you routinely use certain selectors, you can add them to your style sheet and then apply them to individual elements using the Apply Styles panel or by exposing the classes in Intellisense while you're entering them into the tags manually.

11. Click Save and then click Preview on the Common toolbar to view your page in a browser. In the browser view, scroll down to your form and enter some input values to test the validation. Click the Submit button and watch the form validation in action.

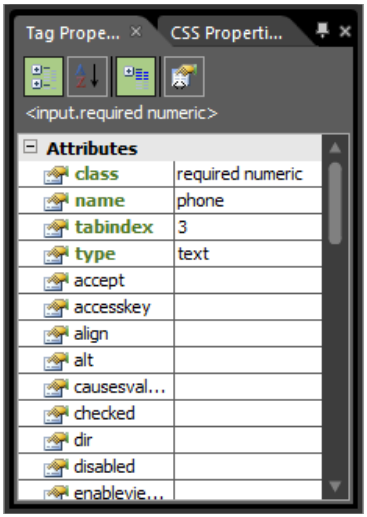


In the image above, I entered nothing into the Name field, entered an invalid email address into the *Email* field, and typed some text into the Phone field. In each case, this input failed to pass the validation function that the jQuery plug-in provides.

12. Close the browser window and return to Expression Web.

13. Select the Phone input field in the Design pane.

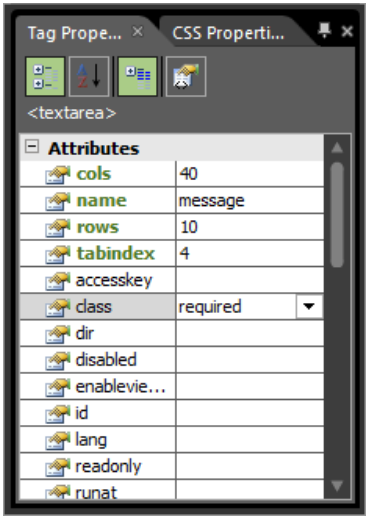
    In many cases, designers will need to modify and experiment with the jQuery, JavaScript, or other attributes they add to their page. Expression Web provides tools that make this task quick, easy, and accurate. The Tag Properties panel is ideal for this operation.



14. With the form field still selected, notice the Tag Properties panel on the lower left of your workspace. Using the Tag Properties panel, remove the class *(required numeric)* from the form field's Tag Properties.

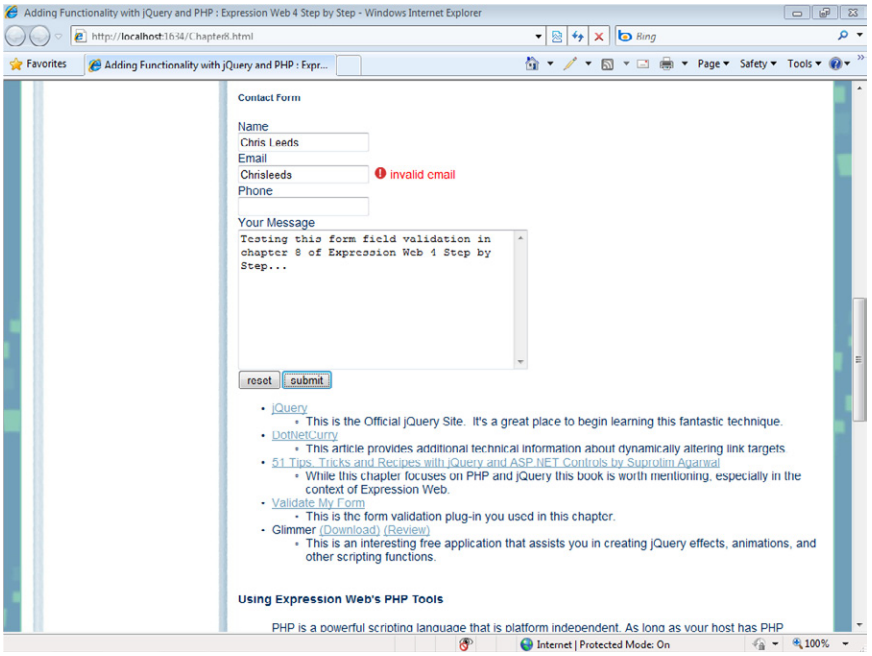
15. In the Design pane, select the message text box of your form and, using the Tag Properties panel, enter **required** in the Class field.

    The idea is to require users to enter something in the message field, but not the phone field.



Tip Use the Set Properties On Top button at the top of the Tag Properties panel to make working with tags much easier. Any attribute that exists for the tag will be at the top of the list.

16. Click Save, and then click Preview on the Common toolbar to check your page in a browser again. This time, test the validation by leaving the phone field blank. Submit the form. You should see a screen similar to the following one.



17. Close the browser window and return to Expression Web.

Although some HTML editors produce JavaScript code, such as the JavaScript that Expression Web created for the Interactive Buttons you used in the media player segment, the HTML editor is best suited for editing the code produced by JavaScript tools such as jQuery and its myriad of plug-ins. By using an open and pre-coded approach such as jQuery, you can make the latest and greatest scripting solutions available in your pages, without having to be a programmer. Using code libraries such as jQuery within Expression Web gives you virtually unlimited client-side scripting capabilities.

Tip Check below the Resources heading in Chapter8.html for links pertinent to this section.

Note Leave the SampleSite open if you are proceeding directly to the next section.

Using the Expression Web PHP Tools

PHP is a powerful, platform-independent server-side scripting language. As long as your server has PHP installed, you can use it regardless of the server type (such as Linux or Windows). When a browser requests a PHP page, the server executes the PHP script and sends the resulting HTML and other content to the browser. PHP pages may serve text, HTML, and script blocks, and function similarly to "Legacy ASP" pages. In other words, PHP pages can mix PHP code with HTML markup and other content.

Microsoft Expression Web 4 provides several tools to make working with PHP easier, and also provides assistance in creating and previewing PHP files. Programming in PHP is a very broad topic, and is well outside the scope of this book. This section serves purely as an introduction to the PHP tools in Expression Web.

In this portion of the chapter, you will become familiar with the Expression Web 4 tools for PHP and how to use them. You will use PHP to create a list of all the images in a folder automatically. You will also learn how to work with PHP functions, variables, and includes in Expression Web 4. At the end of the chapter, you will find references to more information about programming in PHP.

The PHP coverage in this book is not intended to be an example of best practices, or current PHP coding standards. It is here to expose the reader to the PHP tools available in Expression Web, particularly the Insert menu items, IntelliSense, and the Expression Development Server.

Important Before you can complete the rest of the chapter, you must install PHP for the Expression Development Server. See Chapter 2, "Capitalizing on Expression Web 4 Functionality." You will also find some PHP information in Chapter 3, "Capitalizing on the Template Options in Expression Web 4" as it relates to using PHP in template scenarios.

Using PHP in Expression Web

1. From the File menu, select New, and then select PHP. Expression Web creates a new PHP file named Untitled_1.php in your workspace.

2. From the Format menu, select Dynamic Web Template, and then select Attach Dynamic Web Template.

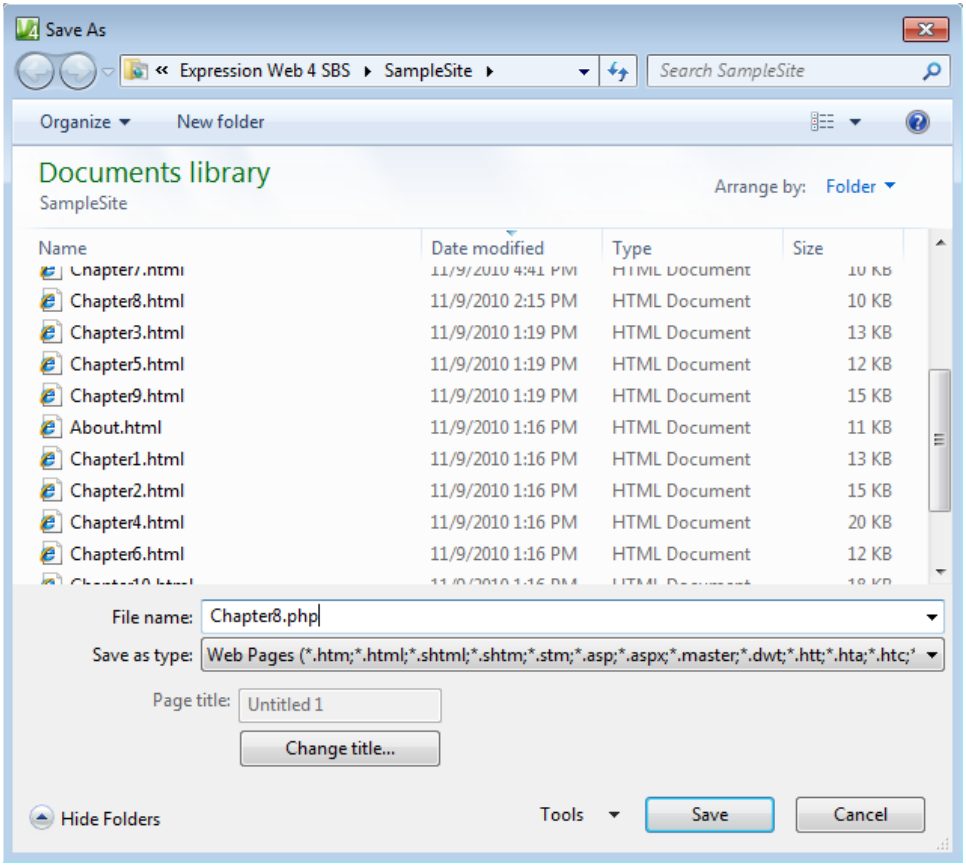
3. In the Attach Dynamic Web Template dialog box, browse to the site's Images folder and double-click main.dwt. Click Close on the Update Confirmation alert.

The method of adding a Dynamic Web Template to a PHP file was originally shown in Chapter 3, so it may seem familiar to you.

4. In the Design pane, set your cursor inside the Content editable region. On the *<DWT:editable>* tab on the Quick Tag Selector, click the drop-down arrow, click Select Tag Contents, and then press Delete on your keyboard.

5. Type **Automatic Image Gallery** and then select the text. On the Styles drop-down menu, click **Heading 1 <h1>**, and then press Enter on your keyboard to create a new paragraph below your heading.

6. Click Save on the Common toolbar. In the Save As dialog box, make sure the root folder of the site is selected, and type **Chapter8.php** into the File Name field.

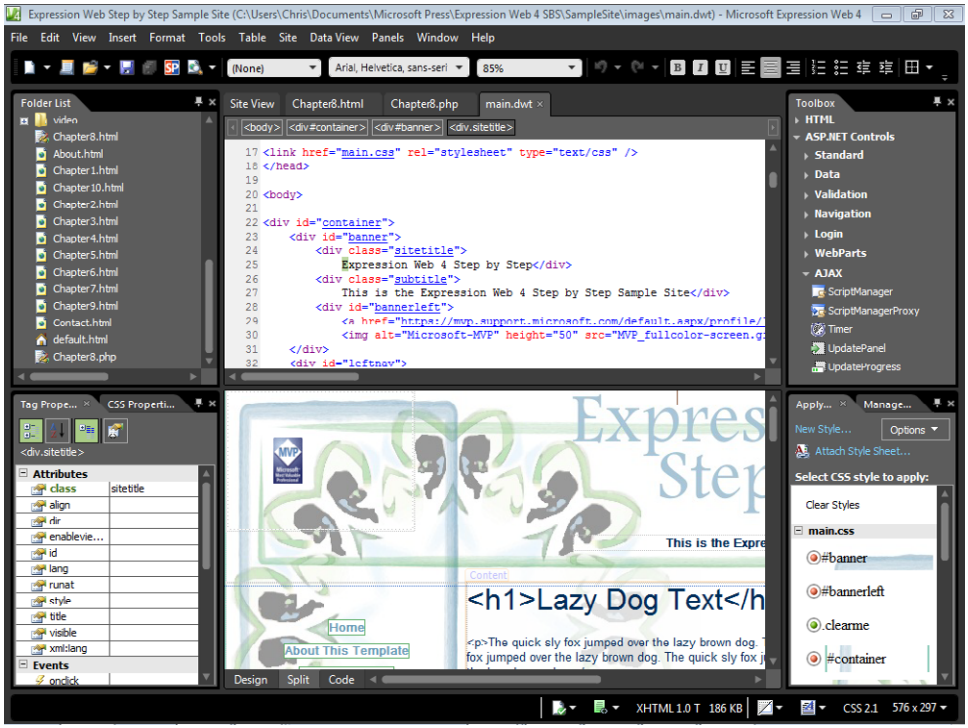


7. Click Save in the Save As dialog box.

   Now that the PHP file is attached to the site's Dynamic Web Template and saved, you need to add it to the site navigation by editing the Dynamic Web Template.

8. From the Format menu, select Dynamic Web Template, and then select Open Attached Dynamic Web Template.

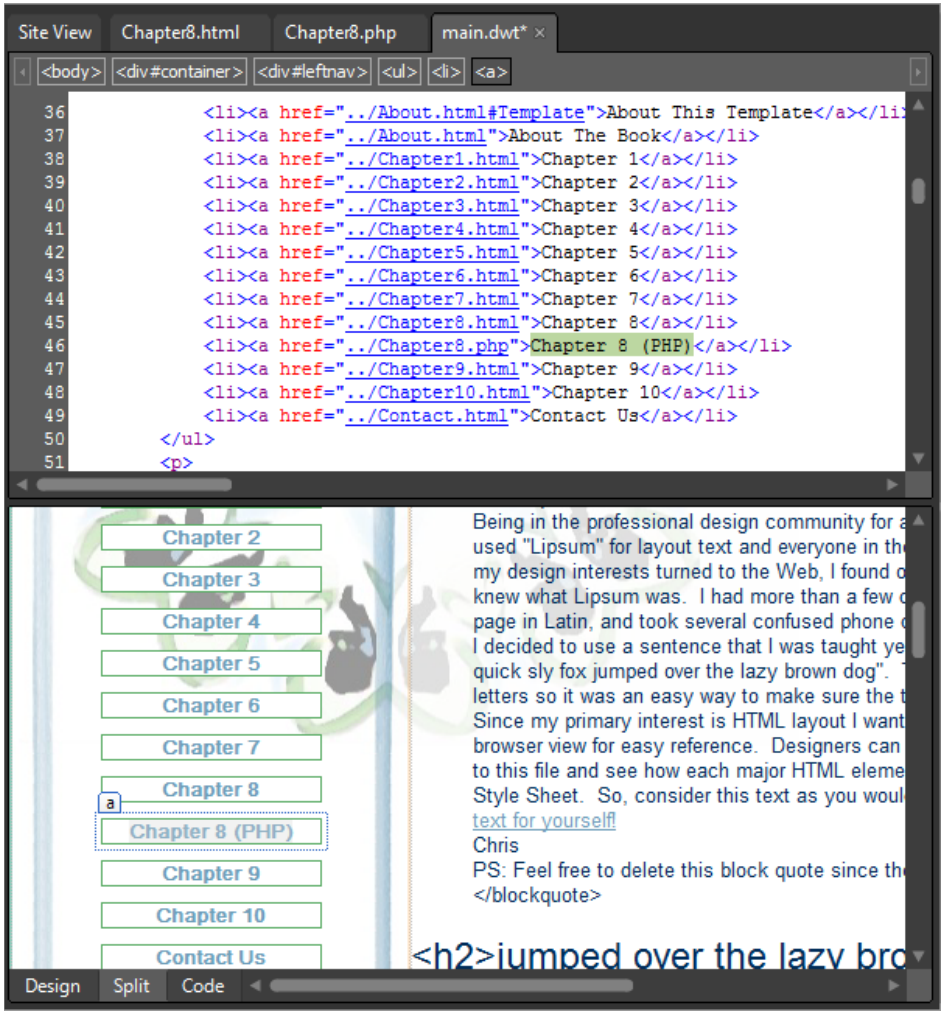
   The file main.dwt opens for editing in your workspace.



9. Set your cursor at the end of the text in the list item containing the link to the Chapter 8 file, and then press Enter on your keyboard to insert a new list item.

10. Type **Chapter 8 (PHP)** in the list item, and then triple-click the text to select it. Right-click the selected list item text, and from the context menu, select Hyperlink.

11. In the Insert Hyperlink dialog box, click Chapter8.php and then click OK to close the dialog box and insert the link.



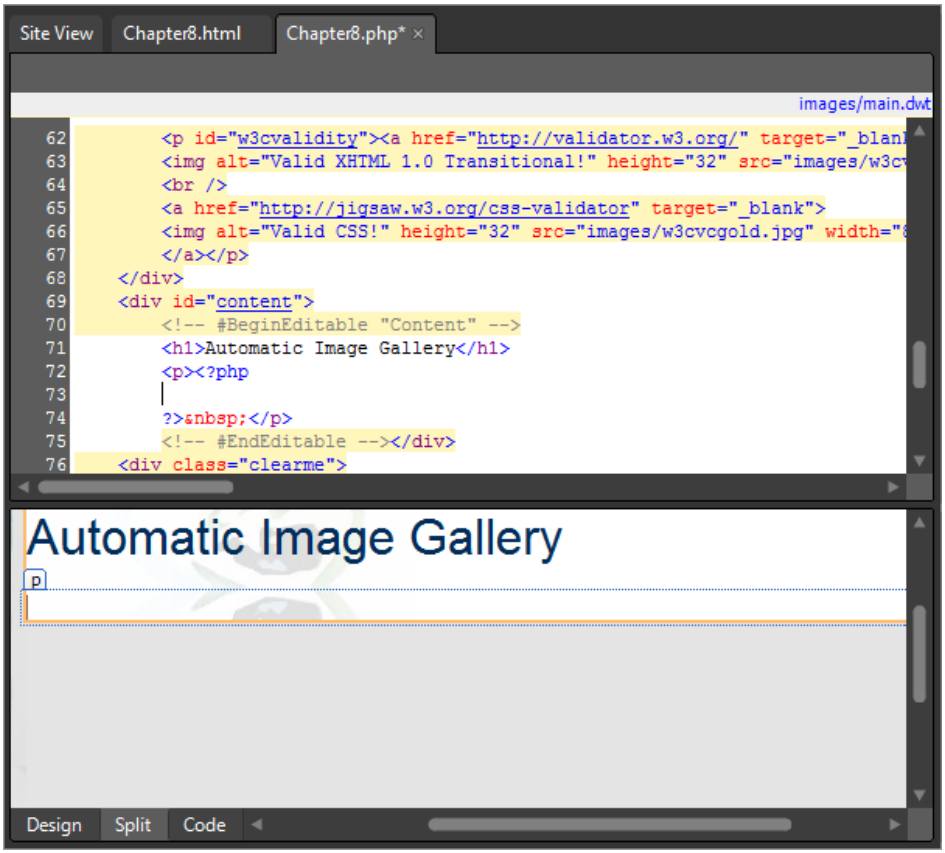
12. Click Save All on the Common toolbar, click Yes on the File Update alert to save your navigation change to the DWT and all the pages it's attached to, and then click Close on the Confirmation alert.

13. Close the Dynamic Web Template and return to Chapter8.php.

Notice how the navigation change you made in the DWT is now present on the new PHP file. The usefulness of a template for maintaining and managing a Web site can't be overstated.

14. In the Design pane, set your cursor in the empty paragraph below the heading that you previously entered, and from the Insert menu, point to PHP and then click Code Block.

15. Set your cursor between the PHP tags in the Code pane, and press Enter twice on your keyboard to create an empty line. Then move your cursor to the new empty line between the PHP tags.



In the next few steps, you will use PHP to create an unordered list of all the files in a particular folder.

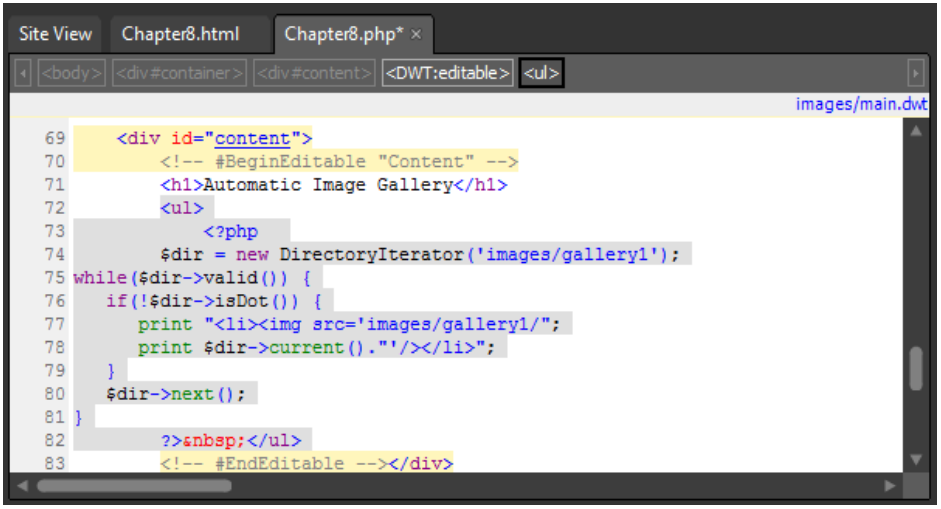
16. Enter the following code:

```
$dir = new DirectoryIterator('images/gallery1');
while($dir->valid()) {
   if(!$dir->isDot()) {
      print "<li><img src='images/gallery1/";
      print $dir->current()."'/></li>";
   }
   $dir->next();
}
```

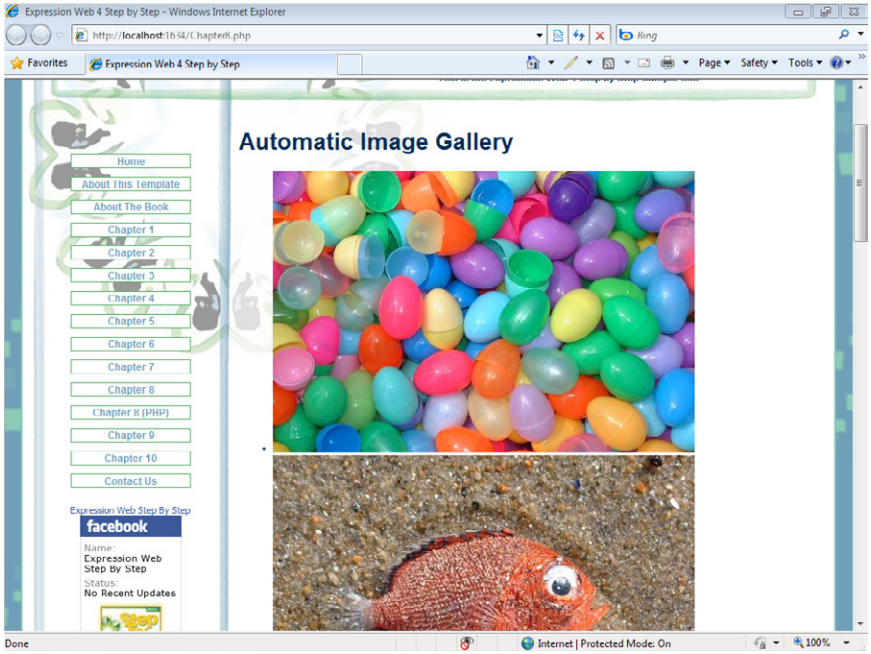
Tip You don't have to type those lines if you don't want to. You can copy and paste them from the file DirectoryIterator1.txt in the site's /files folder.

The code retrieves all the files in the images/gallery1/ folder, and creates a set of HTML list items (*
), to display those images, skipping the two "dot" directory items (the directory itself and its parent) that appear in all child directory listings. The code creates the list items and image tags, but you need to add the unordered list (<ul>*) tag that surrounds the items to make the list render properly.

17. On the Quick Tag Selector, click the drop-down arrow on the *<p>* tag that's surrounding your PHP code, and then click Edit Tag. On the Quick Tag Editor, change the *<p>* tag to a *<ul>* tag and then click the Checkmark button.



18. Click Save and then click Preview on the Common toolbar to run the PHP code and preview your page in a browser.



Troubleshooting For the Expression Development Server to process PHP code, you must install PHP and set the path in either the Expression Web Application or Site Settings. See Chapter 2 for more details.

Take a few minutes to examine the page in a browser. Consider what you've been able to do with just a few lines of PHP. You have created a script that gets the file name of every file in a specific folder, and then displays them as a list of images. If you were to add additional image files to the /images/Gallery1/ folder, those images would also be included in the list; in other words, the list updates itself automatically.

19. Close the browser window and return to Chapter8.php in Expression Web.

Although the PHP worked, there are a couple of things that are sub-optimal. First, if you scroll through the list, you will see a broken image icon. This is because the PHP script looks for all the files in the Image folder, and one of those files is a hidden metadata folder that Expression Web uses to manage the site. PHP tries to display that directory in an image tag, which of course doesn't work. Secondly, the bulleted list is hardly a "gallery." In the next few steps you will remedy both those issues.

Improve the image list

1. Find your PHP code block from the previous steps in the Code pane. Change the code so that it looks like the following:

```
$dir = new DirectoryIterator('images/gallery1');
$FileType=".jpg";
while($dir->valid()) {
   if(!$dir->isDot()) {
      if(strpos($dir->current(),$FileType)) {
         print "<li><img src='images/gallery1/";
         print $dir->current()."'/></li>";
      }
   }
   $dir->next();
}
```

   What you've done is added a variable to contain the file extension ".jpg" (the first bold line in the preceding code) and an *if* statement that makes sure the file name contains ".jpg" (the second bold line). The *if* test skips any file name that doesn't include the extension ".jpg"—in other words, non-image files.

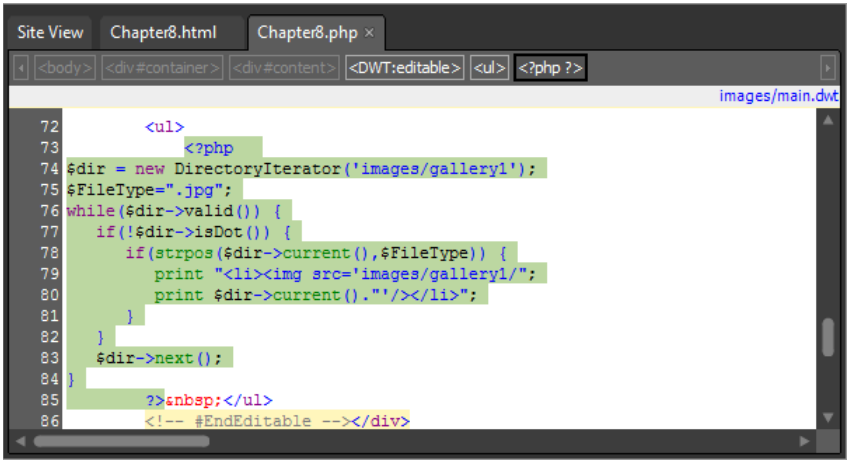
2. Click Save and then click Preview on the Common toolbar to check your modifications in a browser.

   Notice that the previous broken image list item is gone. The list now includes only files with .jpg extensions, which is ideal for this particular gallery.

3. Close the browser and return to Expression Web.

   In the next few steps, you will use more of the Expression Web PHP tools to make the code you've been working with more useful and reusable as well.

4. In the Code pane, locate the PHP code you have been working on. Select all of it (including the opening PHP delimiter "<*?php*" and the closing delimiter "?>").



5. With the entire PHP block selected in the Code pane, right-click the highlighted code, and select Cut from the context menu.

   In the next few steps, you will move this PHP code from Chapter8.php to its own file, and then bring it back into play via a PHP include.

6. From the File menu, select New, and then select PHP. Expression Web creates a new PHP file in your workspace.

7. Set your cursor in the Code pane of this new page, and press Ctrl+A on your keyboard to select all of the contents, and then press Delete on your keyboard.

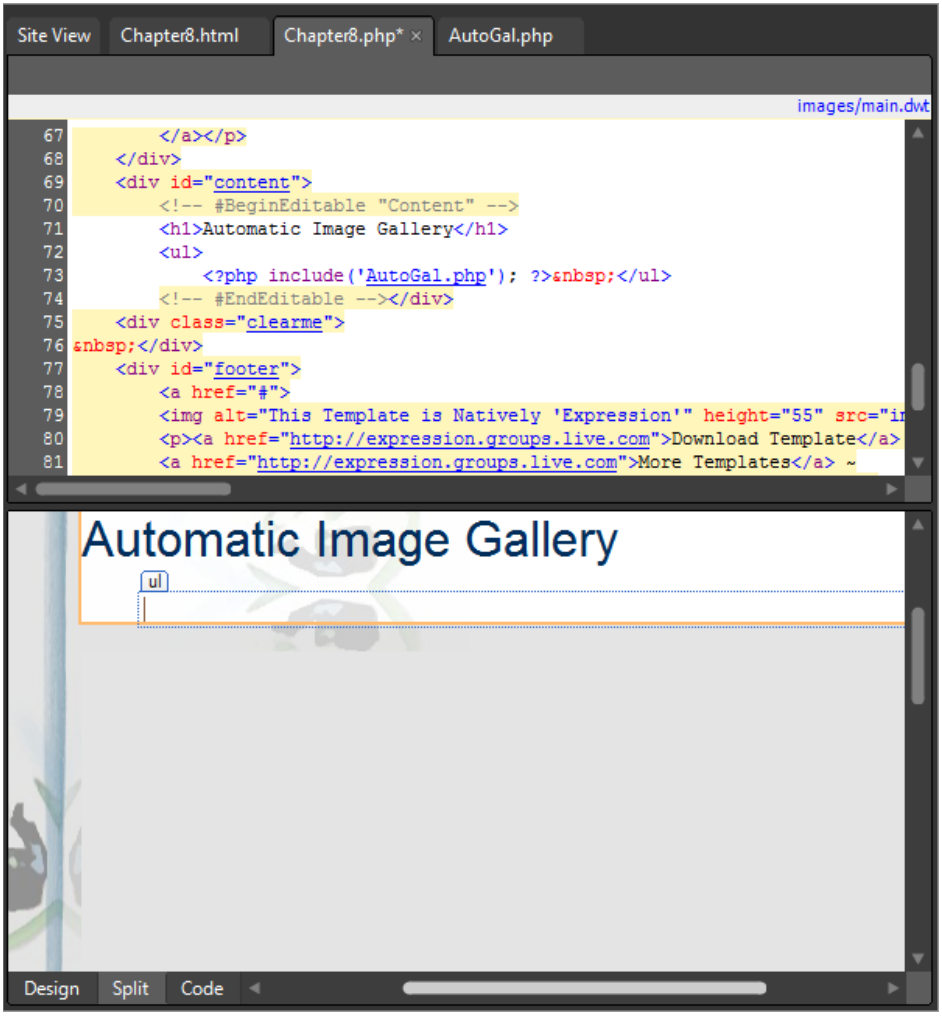
   Because you are creating a PHP file that will be included into a different page, you don't want any head, body, or other tags to be present in it.

8. With the new PHP file completely empty, press Ctrl+V on your keyboard to paste in the PHP block that you cut from Chapter8.php.

9. Click Save on the Common toolbar, and in the Save As dialog box, name this file **AutoGal.php**. Save it in the root of the sample site.

10. Click the Chapter8.php tab at the top of your workspace to continue editing that file.

11. In the Code pane, set your cursor between the *<ul>* tags where you cut the PHP code block from the page. From the Insert menu, select PHP, and then click Include. In the Select File To Include dialog box, double-click the AutoGal.php file you saved in step 9.



The content from AutoGal.php will now be included in Chapter8.php via a server-side include. This is helpful, because you can use an include file on any page you want, and if you need to modify the file, you only need to make the modifications in one location.

12. Click the AutoGal.php tab at the top of your workspace to continue editing that file.

Remember what this code does: it looks in a specific folder, retrieves all the files in that folder, and then filters the list for just the .jpg files, creating an *<img>* tag for each .jpg file, and surrounding each item with *
* tags. So, you could insert it into any file you like but it would always do the same thing. That's not very flexible. In the next few steps, you will modify this file so it's much more flexible in a designer's workflow.

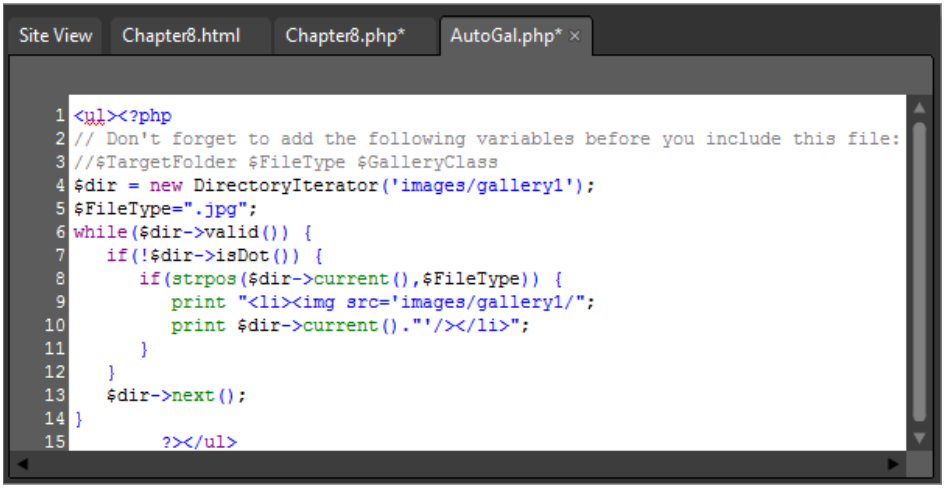
13. The code returns the images as list items. Remember, you had to add the unordered list *
* tag to surround the list before. To make things a little more sensible, you can add the unordered list tags to this include file. Enter a *
* at the very beginning of the file, and a closing *
* at the very end of the file.

14. Set your cursor after the opening PHP delimiter *<?php*, and then press Enter to create a new line.

    Because you will replace some of the hard-coded variables in this block with variables you will be able to set in the page it'll be included on, it's important to make some notes here so that you and or anyone else will be able to easily see what's going on.

15. On the new line just below your opening PHP tag, type the following comment (comments begin with two forward slashes in PHP).

```
// Don't forget to add the following variables before you include this file:
//$TargetFolder $FileType $GalleryClass
```



    By using the PHP comments feature (*//*), you can leave a message in the code file. The PHP engine ignores comment lines when it processes the page, so the comments will never appear in a browser, even if users view the HTML source of the page.

16. Change the *<ul>* tag at the top of this file to:

```
<ul class="<?php echo $GalleryClass ?>">
```

    The *echo* command causes PHP to insert the contents of the following variable (in this case *$GalleryClass*). Because you can set this variable from outside the include file, that means you can define the unordered list class that's in the host file. This will make it easier to apply any gallery style you like.

17. Change the next line from:

```
$dir = new DirectoryIterator('images/gallery1');
```

to:

```
$dir = new DirectoryIterator($TargetFolder);
```

You're using a variable to set the script's path to an appropriate folder. This way, you'll be able to point the script at any folder you like from the page that contains the calls to the include file.

18. Change the following line from:

```
print "<li><img src='images/gallery1/";
```

to:

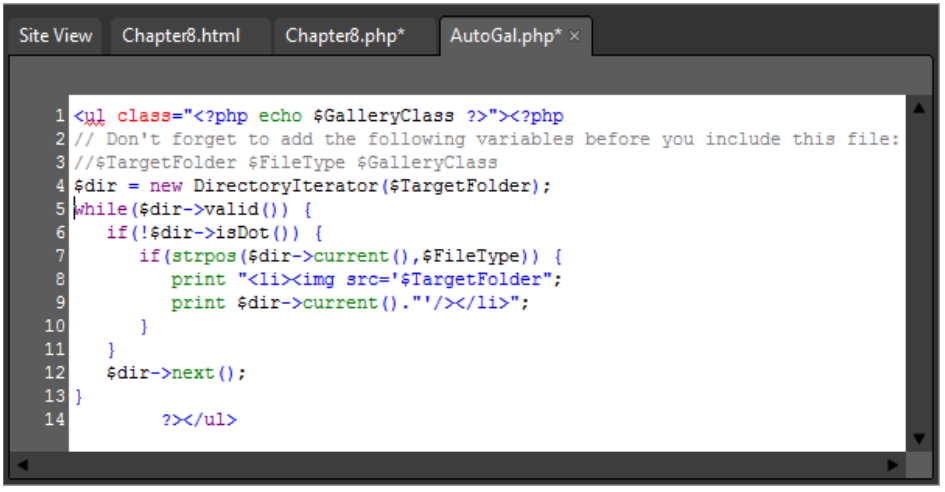
```
print "<li><img src='$TargetFolder";
```

Because you created a variable for the folder path, you want to replace the hard-coded path with the variable name.

19. Finally, remove the following line:

```
$FileType=".jpg";
```

You will define which file type to filter for in the Chapter8.php file, so it doesn't belong in the include file now.



20. Click the Chapter8.php tab at the top of your workspace to switch back to that file. Scroll the Code pane to the *<head>* section and set your cursor just before the closing *<!-- #EndEditable -->* in the "doctitle" editable region, then press Enter on your keyboard to break to a new line. Enter the following code using the Expression Web IntelliSense pop-up:

```
<script type="text/javascript" src=
  "/files/jquery-ulslide/js/jquery.js"></script>

<script type="text/javascript" src=
  "/files/jquery-ulslide/js/jquery.mousewheel.js"></script>

<script type="text/javascript" src=
  "/files/jquery-ulslide/js/jquery.ulslide.js"></script>

<script type="text/javascript">
  $(function() {
     $('.slide1').ulslide({
       width: 443,
       height: 'auto',
       bnext: '#e1_next',
       bprev: '#e1_prev',
       axis: 'y',
            mousewheel: true,
            autoslide: 3000});

       $('.slide2').ulslide({
          width: 443,
          height: 300,
          duration: 1000,
          affect: 'fade',
          bnext: '#e2_next',
          bprev: '#e2_prev',
          direction: 'f',
          duration: 350,
          autoslide: 3000});
  });
</script>
```

Tip You can copy and paste this content from DirectoryIterator1.txt in the site's /files folder.

The three *<script>* tags at the beginning of this code are includes for a jQuery library file and a plug-in called ULSlide. The actual JavaScript below that defines the appearance and behavior of the gallery. You can use other plug-ins that build a gallery based on an unordered list, or you can modify the PHP to create the chosen output that the plug-in requires.

Tip If you are going to use jQuery or some other script in all or many pages, you can include a link to it in your Dynamic Web Template. That way, you would only have to call the plug-in on specific pages because the main library file would automatically be included on all pages.

21. Scroll down through the Code pane until you reach the area in the Content region where you have the PHP Include.

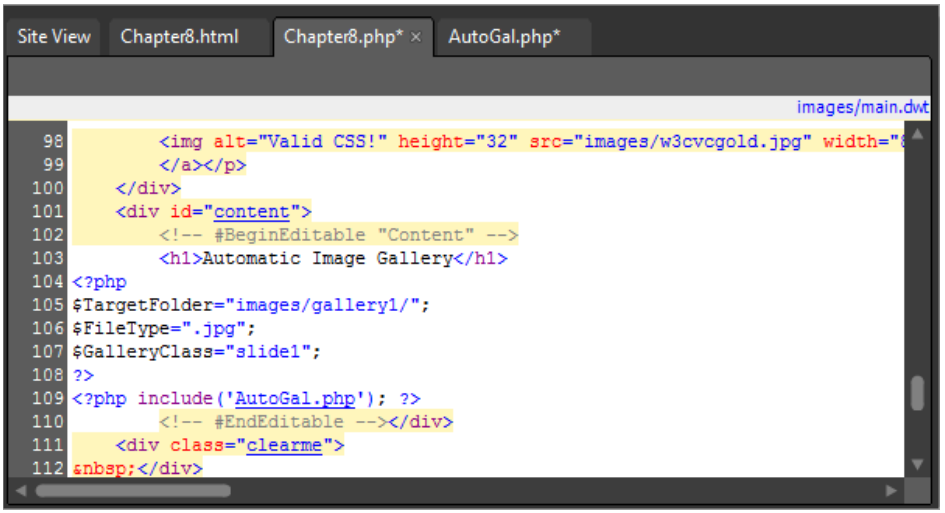
The final work here requires defining a few variables and removing the *<ul>* remnants from the earlier version.

22. Remove both the opening *
* tag before the PHP include and the closing *
* tag after it.

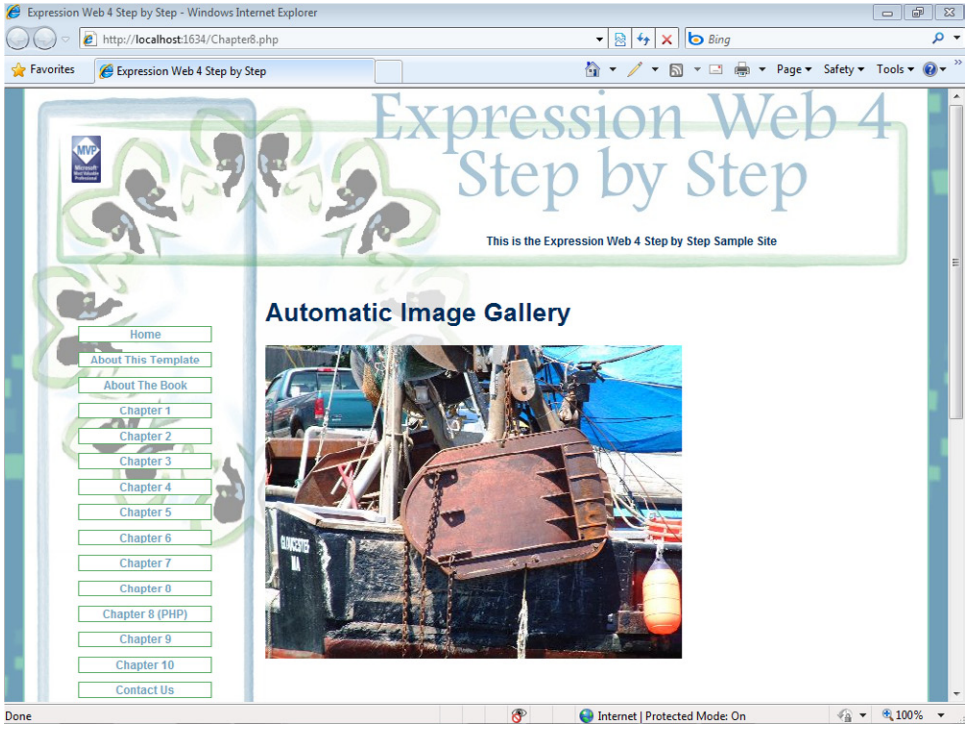
Note You moved those tags to AutoGal.php, so leaving them in place in the Chapter8.php file would create duplicate HTML tags, which would cause rendering and validity problems.

23. Enter the following code just above the PHP include file:

```
<?php
$TargetFolder="images/gallery1/";
$FileType=".jpg";
$GalleryClass="slide1";
?>
```



24. Click the Save All button, and then click the Preview button on the Common toolbar to test the gallery in a browser.



By using a small amount of PHP code and an easy-to-use jQuery plug-in, you have created a system that you can use over and over on any page or any number of pages you like. You've made it easy to turn a folder full of images into a self-maintaining image gallery of sorts.

25. Close the browser window and return to Expression Web.

Most of the work you just completed on the AutoGal.php include file was intended to make it more flexible. To explore that flexibility, you will now change the variables in Chapter8.php and cause the script to get a different kind of file, from a different folder, and apply a different gallery treatment to them.

Explore the power of generic include files

1. Return to the Chapter8.php file. Locate the variables you added to your page just above the PHP include statement and change them from this:

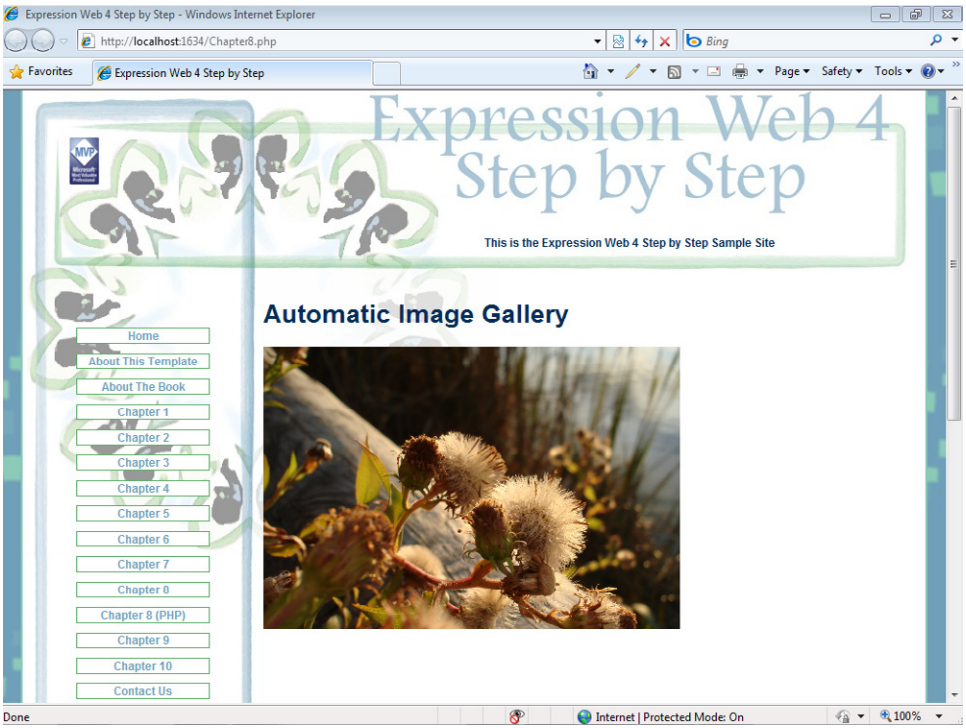
```
<?php
$TargetFolder="images/gallery1/";
$FileType=".jpg";
$GalleryClass="slide1";
?>
```

to this:

```
<?php
   $TargetFolder="images/gallery2/";
   $FileType=".png";
   $GalleryClass="slide2";
?>
```

These new variables tell the script to look into a different folder (images/gallery2/), for a different type of picture file (.png), and then (by changing the $GalleryClass class name), apply a different style to the result.

2. Click Save and then click Preview on the Common toolbar to check your changes in a browser.



As you can see, by changing the variables that pertained to the folder you wanted to use, the file type that you wanted to find, and the jQuery effect you wanted to apply, you were able to include a totally different gallery into Chapter8.php. Using PHP variables and an include file makes it easy and flexible to drop a gallery on any page you like.

3. Close the browser and return to Expression Web.

Note At this point, you can close the SampleSite site and any open browsers. If you are not continuing directly to the next chapter, exit Expression Web 4.

Tip Check below the Resources heading in Chapter8.html for links pertinent to this section.

PHP isn't necessarily a "programmers only" tool. Like ASP.NET or any other server-side scripting, with a little knowledge and some creative use, designers can save a great deal of time, and do some truly interesting things automatically.

By using server-side code appropriately, you can create almost anything a site requires. Knowing how technologies work together helps a great deal. As you've seen in this section, combining PHP with a little jQuery can create interesting and useful results.

Key Points

- Using jQuery in Expression Web can help you enhance your pages with client-side functionality.
- Using the Expression Web PHP Tools can help you add server-side functionality to your pages.
- jQuery plug-ins add tremendous functionality.
- Designers can use jQuery and PHP code together to create interesting and reusable functionality.

Index

Symbols

// (double slash), in PHP comments 373
@ Register directive 148
/ (slash), in FTP location path for site import 460
_ (underscore), files and folders beginning with 475

A

Access Board Section 508 244
Access databases. *See* Microsoft Office Access databases
accessibility 231
  testing, file DOCTYPE and 237
  verifying and addressing 244–247
Accessibility Checker dialog box 244–247
  Check Where, Check For, and Show options 244
  meaningful text in image alt attribute 246
Accessibility panel 245
Accessibility Properties dialog box 189, 253
  exported file from Expression Design 203
  file import from Photoshop 211
Accessibility Reports (Tools menu) 244
action bar (Expression Design) 199
  dimensions and placement of selections 201
  Registration Point button 201
Action field (Options for Custom Form Handler dialog) 348
Action heading (Save Embedded Files dialog) 193
active content 323
active state (links) 289
Actual Size (View menu) 205
Add An XML File (Data Source Library) 332
Add Browser dialog box 39
Add Effects button 262
Add File To Import List dialog box 285
add-ins 101–104
  categories of 101
  more information on 104
  viewing, enabling, and disabling add-in-based components 102–104
Additional Window Sizes (browsers) 40
Address field (Insert Hyperlink dialog) 309
Add Site dialog box 477
Add To Current Site (Import Site Wizard) 460
Add To Managed List (Open Site dialog) 11, 465
Adobe Dreamweaver 95
Adobe Formats 202
Adobe Photoshop files, using in a Web page 209–213
AdRotator controls 144
  inserting and modifying 411–415
  using AJAX to cycle items 430–433
AdRotator Tasks pop-up, Choose Data Source 412
<Ad> tags 413
<Advertisements> tags 413
AJAX controls (ASP.NET) 382
  cycling items in AdRotator without refreshing page 430–433
All Files report 17
all (media type) 322
alt attribute for images, accessibility requirements 245
alternate text field (Accessibility Properties dialog) 253
Alternate Text field (Accessibility Properties dialog) 189
alternative media, styling for 316–322
anchor pseudo-classes 289
Appearance group (Tag Properties panel) 392
Appearance tab (Picture Properties dialog)
  Size group 189, 193
  Wrapping Style and Layout groups 194
  Wrapping Style group 208
Appearance tab (Properties panel)
  contents of 261
Appearance tab, Properties panel 261
  Fill button 261, 265
  Stroke button 264
Application Options dialog box 55–61
  Configure Editors tab 56–59
  FTP tab 60
  Internet Properties 56
  PHP executable fle 74
  Reports View tab 59
  Startup options 55
Application Options (Tools menu) 55, 74
Apply New Style To Document Selection (New Style dialog) 274
Apply Styles panel 7
  Manage Styles tab 301
  Modify Style 196, 294
  New Style 178, 181, 273
  New Style Copy 183
Artboard (Expression Design) 199
ASCII files 60
ASCX files 149
aspect ratio for pictures 190
  changing for slices 201
  Lock Aspect Ratio option for Silverlight Videos 215
  maintaining while dragging a slice 200
ASP.NET 381–434
  Ajax features 430–433
  converting DWT to master page 383–399
  default file and site navigation 310
  full-scale programming in 383
  groups of controls in Toolbox 382
  linking to data sources and using data controls 415–430
  markup and code in separate files 381
  master page as template for attached content pages 136–145
  master pages 108
  modifying control properties using Tag Properties panel 383

ASP.NET *(continued)*
previewing page through Expression Development Server 65–70
site settings for page previews 50
using AdRotator control 411–415
using site navigation controls 400–411
using Web user controls 145–150
Web user controls 66, 108
ASP.NET Controls (Toolbox) 7
ASPX pages 146
Assume Connection Speed Of field (Reports View, Application Options) 60
AsyncPostBack properties (Timer control) 432
Attach Dynamic Web Template dialog box 111, 126, 363
Attach Style Sheet dialog box 384
external CSS file 301
aural (media type) 322
Authoring tab (Page Editor Options) 62
Byte Order Mark (BOM) settings 64
setting default DOCTYPE and file extension for new files 237
Autodetect (Insert Method label) 223
AutoFormat option (Common Tasks) 411
AutoHide All Panels (Panels menu) 30
AutoHide, applying to a panel 7, 8
automatic image gallery, creating using PHP 362–377
modifying gallery using generic include files 378
AutoThumbnail 63

B

Background category (New Style dialog) 276
fonts in tables 179
setting background color for <body> element 287
styling container 273
styling header 274
backing up server-based sites 459–465
Baseline and Comparison panes, SuperPreview 42–44
Baseline browser selector (SuperPreview) 42
Basic Stroke 262
<b> (bold) tags 173, 248
replacing with semantically meaningful tags 174
Behavior group (Tag Properties), StaticDisplayLevel field 405
Behaviors panel
Insert
Set Text 330
opening 330
binary files 60
bitmap files
image export format 202
resizing and saving in different format 189–192
Block category (New Style dialog) 292, 312
block (Display field) for navigation links 312
Block Selection label 24, 273
enabling 158
BMP files 202
<body> tags
copying and pasting content between documents 121
DWT, selecting, copying, and pasting into master page 385
styling with CSS 286–288
transferring content to DWT 303
Bold button (Common toolbar) 173, 342
bold text
Bold, Font-Weight field 251
semantic markup and 173
BOM (Byte Order Mark) settings 63–65
Book Antiqua font 250
bookmarks
in code 28
Code View toolbar buttons for 27
folder-based navigation and 310
setting up along with hyperlink 294
Border category (New Style dialog) 312
styling for navigation menu links 313
Borders And Shading, choosing for media player control 329
Border Thickness (Picture Properties) 194
Box category (Modify Style dialog) 196
Box category (New Style dialog) 277
styling container 273
styling <h1> tag 293
styling header 274
box model (CSS) 280
styling <body> element 287
styling heading (<h1>) tags 293
braille (media type) 322
breadcrumb navigation 400
broken hyperlinks 18
Browse dialog box 58
Browse For Folder dialog box 203
browsers
applying default padding and margins 287
browser view of pages within Expression Web 29–33
downloading and installing 38
encryption of data in secure sessions 441
execution of client-side scripts 324
filling out and submitting a form 348
installation location 39
previewing pages in 36–41
previewing pages with SuperPreview 41–45
previews from, in Snapshot panel 32
rendering issues caused by lack of DOCTYPE 236
rendering Silverlight projects, MIME types for 228
servers passing pages to 324
built-in FTP publishing feature 449
bulleted lists 163. *See also* lists; unordered lists
changing bullet style 181–183
creating 164–166
matching items with headings in Code view 27–29
Bullets button (Common toolbar) 164
buttons
Interactive Buttons dialog box 327
on toolbars 1
Buttons list (Interactive Buttons dialog) 327
Byte Order Mark (BOM) settings 63–65

C

C# 67, 137
C++ 101
Cascading Style Sheets. *See* CSS

case, modifying for definition list text 185
cells, merging 336
certificates, SSL 439, 449
CGI installation type for PHP 72
Change Title button (Save As dialog) 308
character encodings 51. *See also* encodings
Check For option (Accessibility Checker dialog) 244
Check If Expression Web Is The Default Editor For Pages 55
check page validity procedure 238–243
Check Where options (SEO Checker dialog) 494
Choose A Database dialog box 424
Choose Data Source
  AdRotator Tasks pop-up 412
  GridView Tasks pop-up 416
  ListView Tasks pop-up 424
  Menu Tasks pop-up 404, 407
Choose Editable Region For Content dialog box 126
Chrome 38
classes, CSS 187
Class field (Tag Properties panel) 360
Clear Bookmarks (Code View toolbar) 27
Clear field (New Style dialog, Layout) 277
clickable links, styling and 290
client-side functionality 323–350
  creating and using HTML forms 343–350
  server-side versus client-side scripting 323
  using Data View in HTML page 332–343
  using Layers and Behaviors task panes 324–331
Close All Pages (Window menu) 21
closing slash (/) for XHTML tags 240
code. *See also* Code view
  formatting 63
  separation from markup in ASP.NET 381
Code Bookmark 28
Code pane
  displaying code snippets library 232
  DOCTYPE information 234
code snippets
  Code Snippets tab, Page Editor Options dialog box 63
  displaying list of and opening in Code pane 234
  DOCTYPE declarations 232
Code view 25–29, 75, 109
  Ctrl+/ to insert HTML comment 248
  HTML code in 113
  IntelliSense for PHP 71
  line and column numbers 8
  matching list to h4 elements in page body 27–29
  opening the toolbar 27
  PHP include 154
  Quick Tag Selector 25, 117
  in split view 23
  text (.txt) files in 285
  using with Snapshot panel 30–33
  Web user control 148
Code View toolbar 27
  Insert HTML Comment button 27
  Next Bookmark button 29
  Toggle Bookmarks button 28
colors
  changing background color for a control 326
  changing for heading tags with CSS 288
  choosing color schemes 55
  choosing for graphics template 267
  Color Coding tab, Page Editor Options dialog box 63
  Color Dropper tool 206, 261
  Color Picker tool 199, 261
  Color Sliders 261
  in files imported from Photoshop 210
  fonts 179
  selecting for fonts in New Style dialog box 286, 313
  selecting for use in page template 266
  styling for footers 295
  styling for hovered links 290
  styling for links 289
  styling for navigation link menu 312
columns
  choosing for GridView control 417
  editing in Data View 338
  editing in GridView control 419
  inserting into Data View 336
.com (company or commercial) domain extension 438
commands
  add-ins 101
  menu 1
comments
  CSS or HTML, inserting with Code View toolbar 27
  Insert HTML Comment button, Code View toolbar 27
  inserting HTML comment and pasting image inside tag 392
  PHP 373
  removing in HTML optimization 53
Common Tasks pop-up
  AutoFormat option 411
  Create Custom Content 422
Common toolbar 1, 6
  Bold button 173, 342
  Bullets button 164
  Italic button 175
  New Document button
    ASPX 65
    HTML 67
    Page 121
    PHP 74
  Preview button 36
  Save All 366
  Style list 169
  SuperPreview button 41
Comparison and Baseline panes, SuperPreview 42–44
Comparison browser selector (SuperPreview) 42
compatibility 8
  cross-browser, emphasized by Web standards 231
  file DOCTYPE and 237
  HTML Incompatibility option, Code view 27
Compatibility Checker dialog box 238–244
  All Pages selection 243
  buttons in 239
  Refresh Changed Results button 241
Compatibility Layer 211
Compatibility panel 239
Completed tab (Publishing Status) 447
Complete Word (Code View toolbar) 27
Compositions segment (SuperPreview) 42
Configure Data Source dialog box
  Access database as source for GridView control 416–418
  AdRotator control 412

Configure Editors tab (Application Options dialog) 56–59
Configure Editors tab (Application Options dialog) *(continued)*
Microsoft Expression Design, Make Default 58
Configure ListView dialog box 425
Configure The Select Statement dialog box 424
confirmation page for form input 348
connections
Assume Connection Speed Of field (Reports view) 60
Maximum Simultaneous Connections 443
network and Internet 56
Connection Settings dialog box
file system publishing 454, 479
FTP connections 443–445
HTTP connection 449
Location field 443
Connection Type drop-down list
File System 454, 479
FrontPage Server Extensions 449
in Connection Settings dialog box 443
in Import Site Wizard 460
Connection Type drop-down list (Import Site Wizard) 97
Connect To drop-down list 445
in Publishing view 480
contact form sending user input to results page 343–349
container , laying out 273
Container Name and Location fields (Export dialog) 203
Container Name field (Export dialog box) 269
content 157–230
adding images to Web page 188–201
custom, adding to pages created from master page 396
Deep Zoom Composer projects, using in Web page 222–228
defined 157
editing images with Expression Design 198–210
lists 163–172
MIME types to include in your Web site 228
semantic markup, meaning instead of appearance 173–176
separation from presentation 247
Silverlight videos in a Web page 213–222
styling presentation of 177–189
using Photoshop files in Web page 209–213
using tables 158–163
content , styling 275
positioning 278
content pages
ASP.NET, adding navigation control 401–406
combination with master page in ASP.NET 144
matching editable regions to DWT file 126
ContentPlaceHolder controls 141, 386
adding contents of navigation division to 395
converting DWT editable regions to 387–390
Custom or Master content 402
content placeholders 138
in ASP.NET master page 386
Content Tasks button (Design view) 143
Content Tasks pop-up
Create Custom Content 402, 415
Default to Master's Content 407
context menus 12
controls (ASP.NET) 7
AdRotator 411–415
AJAX 430–434
converting DWT editable regions to ContentPlaceHolder controls 386–390
converting image tags in DWT to image controls 391–395
Data group, in Toolbox 423
GridView data control 415–421
groups of, in Toolbox panel 382
insertion into pages without programming code 381
ListView data control 422–430
modifying properties using Tag Properties panel 383
navigation controls 400–411
requirements for correct display 433
controls (form) 344
Convert To Subsite (folders) 486
Create Custom Content (Common Tasks) 422
Create Custom Content (Content Tasks) 402, 415
Create From Dynamic Web Template 111
Create From Dynamic Web Template (File menu) 467
creating Web sites from scratch 257–322
CSS style sheet for alternative media 316–322
designing site architecture and navigation 297–316
graphics template 257–271
HTML page layout 272–284
styling major HTML elements 284–296
Credentials field (Connection Settings dialog) 443
cropping images 199
cross-browser differences from unstyled HTML elements 287
cryptography, using encryption for published information 441
CSS (Cascading Style Sheets) 63
Apply Styles and Manage Styles panels 7
attaching style sheet to DWT 305
attaching to ASP.NET master page 384
benefits of using 187
box model 280
changing for DWTs 84–91
creating 300
creating HTML page layout 272–284
creating style sheet for print version of pages 316–322
CSS tab, Page Editor Options dialog box 63
for Dynamic Web Templates 80
Insert CSS Comment button (Code View toolbar) 27
media types 322
modern Web design using 8
saving style sheet in Images folder 300
setting navigation menu style 311–313
styling HTML elements 177–187
definition list 185
ordered list elements 183
table <th> tags 177
unordered list elements 181–183
styling major HTML elements 284–296
<body> tags 286–288
changing color for all heading tags 288
footer, bookmark link and styling 294
<h1> tags 292
links 289

 tags 290
styling text 247–256
transferring from HTML file to DWT 122
use for layout 158
using external style sheet 300
CSS Class 187
CSS ID 187
link to ID element serving as bookmark 294
CSS Properties panel 8, 280
changing padding and position for element 281
CSS Schema 8
CSS Styles (Format menu)
attaching style sheet to DWT 384
Attach Style Sheet 301
Manage Style Sheet Links 85, 88
custom preview URL 470
cut and paste, keeping content while modifying parent element 389
cutting and pasting text 249

D

databases
drawing information from and displaying 415
Microsoft Office Access and SQL Server 436
sensitive user information in 441
data controls 382
GridView 415–421
ListView 422–430
Data Source Configuration Wizard
Access Database, for GridView control 416
Access Database, for ListView control 424
Site Map 404
XML File 412
Data Source Details panel 333
Data Source Library panel 332
feed.xml file in 333
Show Data 333
Data Source Properties dialog box 332
data sources
linking to, and displaying data with data controls 415–430
XML, displaying data from 6
Data Tasks button, Edit Columns 338
Data View 6, 332–343
changing behavior of hyperlinks 355
editing source code to change columns and move pubDate 337–339
editing source code to correct rendering of HTML markup 339–341
editing source code to refine appearance 341
inserting into HTM/HTML page 332–336
Data View menu, Insert Data View 332
Data View Properties dialog box, HTML View Styles 343
Data View Tasks pop-up menu 334
Change Layout 343
DAV (Distributed Authoring and Versioning) 440
<dd> (definition) tags 170
Decrease Indent Position button 165, 167
Deep Zoom Collections 223
Deep Zoom Composer Online Help Manual 228
Deep Zoom Composer projects, using in Web page 222–228
resources for learning 227
Deep Zoom Composition 223
default file in the root folder of hosting account 448
Default Fonts tab (Page Editor Options dialog) 63
default.html file 68, 83, 93
copying and making it default file for all folders in root 308
opening in Hyperlinks view 18
saving new site's home page in root folder 308
default.master file
adding navigation control 400
creating page from 408
linking page to navigation menu 397
pasting DWT Meta tags into 388
pasting menu navigation control into 406
saving in site's root folder 385
Default Page Encoding setting 51, 475
default state (links) 289
Default to Master's Content (Content Tasks) 407
Defined Term <dt> drop-down 169
definition lists 168
creating 169–172
<dl> tags 163
modifying appearance with CSS 185
Delete Files button 475
deleting files or folders 13
deleting temporary files 51
deprecated HTML Markup 233
descriptions of pages 315
Design Elements category (Stroke Gallery) 265
Design.exe file 58
.design file extension 59
design-time includes
benefits of using 136
FrontPage Server Extensions 472
design-time template functionality 108
Design view 22, 113
adding hyperlinks 113–116
Block Selection label 273
Data View Tasks pop-up button 334
PHP include 154
rectifying image rendering 195
Resample icon 190
Resize Options icon 193
Table menu 159
triple-clicking in to select entire content of parent element 114
using Visual Aids 33–36
Web user control 148
Destination Folder field (PHP installer) 72
Destination Name for publishing 445
DHTML (Dynamic HTML) tasks 324
dialog boxes. *See also* individual dialog box names referenced throughout
add-ins 101
Directory field
Connection Settings dialog box 443
Import Site Wizard 460
Disable-Output-Escaping property (XSL) 341
disk location publishing. *See* file system publishing
Displayed Columns panel 338
Display field (Layout category, New Style dialog) 292
setting for navigation links 312
Distributed Authoring and Versioning (DAV) 440
 tags
adding editable region in DWT 306

 tags *(continued)*
CSS/HTML layout 272–285
basic padding for page divisions 280–283
Deep Zoom Composer 224
inserting and styling for media area 325–329
Silverlight video 216
 tags versus 291
styling, and access by Behaviors panel 330
<dl> (definition list) tags 163
dll application/x-msdownload MIME type 228
dmg application/octet-stream MIME type 228
DNS settings for hosting space 437
DOCTYPE 8
complete listing of, W3C 233
declarations in code snippets library 232
setting default 236
understanding and changing 233–237
Document Type Declaration. *See* DOCTYPE
domain names, finding and registering 437
using Microsoft Office Live Small Business 439
DOM (Document Object Model) 44
DOM Tree View 44
Do Not Publish command 213
directory items (dot) 368
drawing lines
using Ellipse tool 265
using Pen tool 264
Draw Layer button (Layers panel) 328
Dreamweaver 95
<dt> (defined term) tags 169
Dynamic HTML (DHTML) tasks 324
dynamic pages
default file and navigation links 310
previewing through Expression Development Server 65
Windows and non-Windows types 436
Dynamic Web Templates (DWTs) 85, 107–131
adding CSS style sheet for print version of pages 320
adding editable region 116–120
attach CSS style sheet 305
Attach Dynamic Web Template dialog box 363
attaching to PHP page 150, 363
changing style sheet attached to 86–91
comments 53
converting to master page 383–399
creating and saving in Images folder 302
creating file based on 307
creating new DWT and transferring layout to it 120–128
creating new page and attaching DWT 109–113
creating new page based on 467
creating site from 80–85
editing to add PHP page to site navigation 365–367
linking new page to navigation 113–117
link to jQuery or other scripts to use on specific pages 376
setting up editable regions 305–307
updating all files attached to 313
use in FrontPage 472
using Include Page feature 128–137
Dynamic Web Template toolbar, contents of 305

E

echo command 373
Edges, Guides, and Grid, displaying 259
editable regions
adding to DWT files 124–126
DWT, converting to ASP.NET content placeholders 386–390
matching for DWT and content page 126
setting up in DWT 305–307
Editable Regions dialog box 117
naming and adding editable region 305
wrapping selected div in editable region 124
Edit Browser List dialog box 38
Edit Columns dialog box
Title in the Displayed Columns panel 338
use in setting up custom data views of XML files 338
Edit Columns (GridView Tasks) 419
Edit Hyperlink dialog box 335
editing server-based sites 465
via FTP 469–472
via HTTP (FrontPage Server Extensions) 466–469
working directly on the server 465
editing window 4
Code view 25
Design, Split, and Code options 23
Design view 24
and Folders List panel 12
opening a file in 20
opening multiple files in 21
size information 8
Edit Local Copy Now, Remote Site Editing Options dialog 469
Edit menu 5
Find 340
IntelliSense submenus 27
Paste Text 249
editors
changing Page Editor Options 61–65
code editors for ASP.NET programming 383
Configure Editors tab (Application Options dialog) 56–59
configuring 56–59
Expression Web as default editor 55
template formats for 95
WYSIWYG 33
Edit Tag option (Quick Tag Selector) 25
Edit Tag (Quick Tag Editor) 368
Edit User Control 149
.edu (educational institutions) domain extension 438
Effects button, Properties panel 206
Effects panel 262
Ellipse tool 266
drawing a circular shape 265
Email form field 358
setting up 346
validation of user input 359
embossed (media type) 322
emphasis () tags 173–175
empty site
creating 92
importing files and folders 93–96
Enable Paging and Enable Sorting (GridView Tasks) 420
Encoding dialog box 215
encodings. *See also* Expression Encoder

default for pages 51, 475
selecting for Silverlight video player 218
time required to re-encode Deep Zoom Composer 224
Variable Bit Rate (VBR) 215
encryption 441
Use Encrypted Connection option 450
English (United States) language setting 475
Error Highlighting (Code View) 27
Exclude From Publishing option 481
Export dialog box
Container Name and Location fields 203
Export All 269
exporting images from Expression Design 201, 207
exporting slices 269
exporting Web Packages 489–492
Export Location dialog box 269
Export Web Package dialog box 489–491
Files In Package list 490
Properties 491
Expression Blend 203
Expression Design 157
community tutorials 199
creating graphics template 257–271
.design file extension 59
editing images 199–210
layout of screen 199
resources for learning 271
Expression Development Server 65–71
changing site settings to preview HTML pages 70
previewing dynamic page 65–70
previews of folder-based ASP.NET sites 382
site settings for page previews 50
using PHP with 71–82
working with run-time template features 109
Expression Encoder 157, 213
Online Help Manual 215
resources for further information 222
Expression Web 4
opening multiple instances 458
starting without a site open 10
Expression Web 4 User Guide 494
Extensible Stylesheet Language Transformations (XSLT) 339
Extension field (Open With dialog) 59
extensions, domain name 438
extensions, file. *See* file extensions

F

F1 key for Help 199
F5 key for refreshes 208
Failed, Completed, and Log tabs (Publishing Status) 447
Fields dialog box 419
file extensions 57
changing for images 191
MIME types 228
.psd (Photoshop) 210
setting .htm or.html as default 237
transfer as ASCII or binary 60
File menu 4
Exit command 9
Export 207, 269
Import 5, 93, 285
New
Create From Dynamic Web Template 467
Create From Master Page 396, 408
PHP 371
Open command 10
Page 384
Preview In Browser 38
Save All 302
File Name field (Save As dialog) 364
File Open dialog box 492
file references, broken 19
files
All Files report 17
creating new file from DWT 307
deleting 13
exported from Expression Design 201
hiding special files 51
links, displaying in Hyperlinks view 18
in renamed folders 299
saving to Expression Web from other applications 203
Select File To Include dialog box 372
updating all attached to DWT 313
using PHP to create list of files in a folder 367–369
using Web Package exports to cull 492
Files In Package list (Export Web Package dialog) 490
Files In Site pane (Export Web Package dialog) 489
file system connection, importing site via 97
file system publishing 440, 454–458
publishing selected files 479
File Transfer Protocol. *See* FTP
File Type dialog box 191
Fill button (Properties panel, Appearance) 261
choosing fill 265
Fill Opacity (Properties panel, Appearance) 261
Fill Type/Stroke Type (Properties panel, Appearance) 261
Find And Replace dialog box 340
Find command (Edit menu) 340
Find Matching Brace (Code View toolbar) 27
Find Matching Tag button (Code View toolbar) 27
Firefox
comparing Firefox 3.6.2 to IE 8 preview 42
download site 38
installation location 39
Flash interactive button 330
Float field (New Style dialog, Layout) 278
float for images 193, 195
changing 196, 242
changing in New Style dialog box 253
half to left and half to right 209
setting margins and padding in New Style dialog box 273
flv video/x-flv MIME type 228
folder-based navigation 50, 67–70
Folder List panel 68
deepzooms folder 225
default.html 88
Do Not Publish 213
Exclude From Publishing option 481
Images folder 203
javascript and silverlightvideos folders 218
opening a page from 20
silverlightvideos folder 221
folders
Browse For Folder dialog box 203
common, saving sites in 65
converting subsites to 488
converting to subsites 486
deleting files or folders 13

folders *(continued)*
Destination Folder field for installation folder 72
editing using context menu 12
exporting as Web Packages 489
hiding special files 51
importing into a site 93
Properties dialog box 73
publishing into 436, 443
renaming 299
using PHP to create list of files in 367–369
Folders List panel 8
hierarchical view of all folders and files in site 12
images 12
folder structure and navigation, designing for a site 297–316
adding heading text, titles, and descriptions to pages 314
creating site folders 298
setting up navigation 309–311
styling navigation menu links 311–313
unordered list for navigation menu 298
Folders view
contents of newly imported site 464
default site view 14
open site in 11
using for file and folder tasks 299
workspace with open site 14
Follow Code Hyperlink (Code View toolbar) 27
fonts
accessibility considerations 244
color set in <body> tag 288
colors for 179
Default Fonts tab, Page Editor Options dialog box 63
Font Families tab, Page Editor Options dialog box 63
modifying style for footer element 294
multi-font specification in New Style dialog box 250–252
selecting and styling in New Style dialog box 286
semantic markup 173–176
styling for navigation menu 312
styling for print version of pages 318
styling in definition list 185
styling in table 179
styling in unordered list 181
Font-Weight field (New Style dialog) 312
footer area of graphics template 263
slice to contain graphics 268
footers
inserting hyperlink and styling with CSS 294
linking to top of page 313
styling footer 276–278
foreign-language pages 51
Format menu 6
Behaviors 330
CSS Styles 85, 88, 384
Attach Style Sheet 301
Dynamic Web Template 363, 365
Manage Editable Regions 117, 124
Open Attached Dynamic Web Template 129
Export dialog box 202
Layers 328
formatting marks, displaying 35
Form Controls 7
Form Controls group (Toolbox) 343, 344
Form Field Properties
email field 346
text area field 346
text box 345
Form Properties dialog box 347
options requiring FrontPage Server Extensions 348
forms. *See also* HTML forms
ASP.NET forms page 386
input validation plug-in 356–361
tools to assist in creation of 343
wrapping ASP.NET content placeholder 138
<form> tags
adding ID 357
selecting, copying, and pasting contents into default.master 385
FrontPage 95
FrontPage Server Extensions
design-time includes and DWTs 472
editing server-based sites via HTTP 466–469
HTTP publishing through 440, 449–453
required by options in Form Properties dialog box 348
FTP 439
Application Options dialog box, FTP tab 60
copying server-based site to local folder via 460–465
editing server-based sites via 469–472
over SSH (SFTP) 439
previewing Web sites open through 50
publishing with 441–448
Fuzzy Brush (Stroke Gallery, Ink category) 264
Fuzzy Thick Wet Brush (Stroke Gallery, Watercolor category) 264
FWP files 489

G

gallery styles, Silverlight templates 214
generated HTML, removing 54
Generator and ProgID tags, removing 54
GenericSiteTemplate folder 94
Geometric Flowers with Stars (Stroke Gallery) 265
get and post methods 348
Gif files, opening with Expression Design 58
glow effect for images 206
creating in graphics template 262
Google-hosted jQuery library 353
graphics
creating and working with 257
images considered as 198
graphics template 198
graphics template, creating 257–271
basic shape for page content background 259–262
color choices 267
color sample squares 266
common page areas 262–265
Properties panel, Appearance tab 261
slicing and exporting the images 268–270
Grid, Guides, and Edges, displaying 259
GridView controls 415–421
choosing and configuring data source 416–418
editing columns 419
enabling paging and sorting 420
setting layout 421
GridView Tasks pop-up
Choose Data Source 416

Edit Columns 419
Enable Paging and Enable Sorting 420
groups of pages, opening and editing 21
Guides, Grid, and Edges, displaying 259

H

handheld (media type) 322
HD Photo files (WDP) 202
header area of graphics template 262
slice to contain graphics 268
headers
styling header division
 tags 274
 tags 290
heading tags 176
adding <h1> tags to pages 314
editable regions in <h1> tags of DWT 305
matching bulleted list items to, in Code view 27–29
styling for print version of pages 319
styling <h1> tags 292
styling tags 290–292
styling with CSS 288
head sections of pages
content placeholders in master pages 138
style sheet links and JavaScript links 123
Help
Expression Design 199
Expression Encoder 215, 222
Help menu 6
Hide Files And Folders That Begin With An Underscore Character 51, 475
HiSoftware Cynthia Says Portal 247
Horizontal Margin (Picture Properties) 194
hosting requirements 436
guidelines for choosing server 436
information for setting up hosting space 436
reseller accounts 437
security 441
hosting, setting up, using Microsoft Office Live Small Business 439
hover state (links)
styling hovered links 289
styling navigation links 313
.htm or.html as default file extension 237
HTML 4.01 Transitional DOCTYPE 235
HTML editors. *See also* editors
Expression Web and 47
making Expression Web 4 the default 2
SEO Reports 493–496
working with JavaScript code 361
HTML forms
creating and using 343–349
validating user input using jQuery plug-in 355–359
HTML (Hypertext Markup Language)
comments 392
creating file to use as include page 131
creating new HTML page 67
file for add-in interface 101
generated HTML, removing 54
HTML tags in Toolbox panel 7
instruction in Split view 113
learning, using DOM Tree View and SuperPreview 45
messages on status bar 8
Open As HTML option 58
selecting entire element in Design and Code panes 24
semantic markup 173–176
styling elements with CSS 177–187
styling major elements 284–296
<body> tags 286–288
bookmark link and styling in footer 294
<h1> tags 292
heading tags 288
links 289
 tags 290
styling text with cascading style sheets 247–256
tags in XML files 341
transferring contents of file to DWT file 121–123
using Data View in HTML page 332–343
W3C-valid HTML code 83
working with, in Code view 26–29
HTML Incompatibility (Code View) 27
HTML layout, creating 272–284
CSS box model, padding for page divisions 280–283
styling tags 273–280
HTML Optimization Settings dialog box 52
HTML View Styles (Data View Properties dialog) 343
HTML/XHTML files
DOCTYPE 232–237
verifying and addressing accessibility 244–247
verifying and addressing W3C validity 238–244
HTTP (FrontPage Server Extensions)
editing server-based sites via 465–468
publishing with 440, 449–453
security and 441
HTTPS, publishing via 441
hyperlinks
altering behavior of, using jQuery 352–355
broken 19
changing behavior of in Data View 355
creating and formatting for Data View 334
editing CSS style sheet for print version of pages 319
formatted, inserting selected fields as 334
inserting for site navigation 309–311
linking footer to top of page 294, 313
linking new pages to navigation 113–117
linking new page to master page navigation 397–399
linking page to default.master file 397
linking page to jQuery library 352
linking page to jQuery plug-in 355
linking PHP page to site navigation 365
placing in ContentPlaceHolder control 141
Recalculate Hyperlinks, Tools menu 203, 269
setting, using Ctrl+K keyboard shortcut 398
styling site navigation links 311
Hyperlinks view 14, 18–20

I

<i> (italic) tag 175
ICANN (Internet Corporation for Assigned Names and Numbers) 438
ID attribute, <form> tag 357
IDE (Integrated Development Environment) 47
IDs (CSS) 187
Ignore The Keyboard When Deciding The Encoding Of New Pages 51
image controls, converting HTML image tags to 391–395, 399
image files. *See also* file extensions; images
  default application for opening 58
image gallery, creating using PHP 362–377
  modifying, using generic include files 378
images
  accessibility requirements, alt attribute 245
  adding to a Web page 188–197
  considered as content or graphics 198
  Deep Zoom Composer 222–228
  editing with Expression Design 198–210
  saving CSS style sheet in Images folder 300
  styling with cascading style sheets 253–255
  thumbnail copy of 63
  using Photoshop files in Web page 209–213
Images folder 12
ImageUrl field (Tag Properties, Appearance) 392
Import Adobe Photoshop File dialog box 210
  opening with Update From Source 211
Import dialog box 93–96
  importing file into site 285
  importing folder into Deepzooms folder 225
  importing videos into silverlightvideos folder 221
Import (File menu) 285
Import From Web Package 492
importing files
  Photoshop files, properties for 210
  into a site 93–96
Import Location dialog box 97
Import Site Wizard 96–101
  Connection Type List 97
  copying server-based site to local folder via FTP 460–465
  Location field 97
Import Web Package dialog box 492
Include command 372
include files
  automatic image gallery created using PHP 362–377
  generic, exploring power of 378
  PHP auto gallery file 372
include methods 108
Include Page button (Standard toolbar) 133, 225
Include Page feature 107, 128–137
Include Page Properties dialog box 133, 226
Include Subsites (Publishing tab, Site Settings dialog) 488
Increase Indent Position button 164, 167
indentation
  Decrease Indent Position button 165, 167
  Increase Indent Position button 164, 167
.ini file (PHP) 74
  changes to 75
Ink category, Fuzzy Brush (Stroke Gallery) 264
input fields of form, validating using jQuery plug-in 355–361
Input (Reset), Toolbox panel 346
Input (Submit), Toolbox panel 346
Input (Text), Toolbox panel 345
Insert CSS Comment button (Code View toolbar) 27
Insert Data View command 332
Insert Deep Zoom dialog box 223
  Cancel button 223
Insert End Tag button (Code View toolbar) 27
Insert HTML Comment button (Code View toolbar) 27
Insert Hyperlink dialog box 114
  linking pages to site navigation 309
  linking page to master page navigation list 397
  linking PHP page to DWT file 366
Insert Layer button (Layers panel) 328
Insert menu 5
  adding picture to Web page 188
  HTML 328
  Interactive Button 327
  Media 5, 213
    Deep Zoom 223
  PHP
    Code Block 367
    Include 153, 372
  Picture, From Adobe Photoshop 210
  Picture, From File 188
  Set Text 330
  Symbol 102
Insert Selected Fields As button
  Formatted, Hyperlink 334
  Single Item View 333
  Sub View 336
Insert Silverlight Video dialog box 214
Insert Start Tag button (Code View toolbar) 27
Insert Symbol dialog box 102
Insert Table dialog box 159
  styles for tables 178
Insert Table (Table menu) 159
installation folder location
  for browsers 39
  for PHP 72
Integrated Development Environment (IDE) 47
IntelliSense
  accessing via Code View toolbar 27
  for CSS 318
  file DOCTYPE and 237
  for jQuery 353
  for PHP 71
  Page Editor Options dialog box 63
  using with PHP code 375
interactive buttons 324–331
  Flash 330
  removing attributes 54
  Windows Media 331
Interactive Buttons dialog box 327
interface objects. *See also* user interface
  menus, toolbars, and panels 1
Internet Corporation for Assigned Names and Numbers (ICANN) 438
Internet Explorer (IE)
  comparing IE8 with Mozilla Firefox 3.6.2 preview 42
  previewing page in Snapshot panel 32
  Print Preview (File menu) 321

Internet Properties dialog box 56
Interval field (Timer control) 431
italic text
 (emphasized) tag 175
Italic button on the Common toolbar 175
Items To Export field (Export dialog) 269

J

JavaScript
client-side scripting 323
disabled in browsers 324
jQuery library. *See* jQuery
JavaScript files
for add-ins 101
links in head section of page source 123, 140
Silverlight.js 219
JPEG files
changing bitmap file to 192
exporting image slice as 202
importing Photoshop images as 210
opening with Expression Design 58
slice export format 268
.jpg files
adding variable to PHP script to filter for 370
opening with Expression Design 58
PHP script filtering image list for 372
jQuery 351–360
linking a page to the library 352–354
linking to plug-in and using it to validate input 355–360
plug-ins 351
using plug-ins with PHP 375
writing a script into the page 353

K

Keep Aspect Ratio and Specify Size (Picture Properties dialog) 190
keyboard, ignoring when deciding encoding of pages 51
keyboard shortcuts 1
copying, Ctrl+C 286
creating line breaks, Shift+Enter 294
cut (Ctrl+X) and paste (Ctrl+V) 286
cycling through open pages, Control+Tab 21
Help, F1 199
increasing/decreasing indentation levels in lists 165
inserting HTML comments, Ctrl+/ 392
refreshing page in workspace, F5 424
selecting all text, Ctrl+A 266
setting hyperlinks, Ctrl+K 398
keywords. *See also* search engine results
entering in Page Properties dialog box 315

L

Language Settings group
Advanced tab, Server Settings dialog box 475
Advanced tab, Site Settings dialog box 51
layers
Compatibility Layer (.psd file) 211
excluding file imported from Photoshop 209
in Photoshop files, including/excluding 210
Layers and Behaviors task panes 324–331
Behaviors panel, setting text for layer 330
Layers panel, Modify ID 329
opening Layers panel 328
Layers (Expression Design) 199
layout
creating an HTML layout 272–284
returning workspace to default 9
use of tables for 158
Layout category (Modify Style dialog) 196
Layout category (New Style dialog) 277
Float field, Right 254
Float field, setting for content 278
navigation menu, Display field 312
setting Overflow and Display fields for header division 292
Layout group (Picture Properties) 194
Layout group (Tag Properties)
image control width and height 393
Orientation field 405
Width field 421
Layout tab (Data View Properties dialog) 343
Learn More About SEO link 494
line breaks
creating, using Shift+Enter 294
lines, drawing
using Ellipse tool 265
using Pen tool 264
links. *See also* hyperlinks
adding to new page 113–117
styling with CSS 289
Link Style Sheet dialog box 87
List category (New Style dialog) 311
List-Style-Type to Lower-Alpha 184
List-Style-Type to Square 181
list item (<li>) tags
displaying images in automatic gallery PHP file 368
in ordered (numbered) lists 167
in unordered (bulleted) list 164
 tag within 166
List Members (Code View toolbar) 27
lists 163–172
creating definition lists 169–172
creating ordered list 167
creating unordered list 164–166
improving image list created by PHP script 370–376
styling definition list with CSS 185
styling ordered list with CSS 183
styling site navigation menu 311
styling unordered list with CSS 181–183
types of lists and their uses 164
using PHP to create unordered list of files in folder 367–369
working with, in Code view 27–29
ListView controls 422–430
choosing and configuring data source 424
configuring 425
editing source code to eliminate text and change layout 427–429
ListView Tasks pop-up
Choose Data Source 424
Configure ListView 425
Current View, changing to ItemTemplate 427
Local Copy Location field (Import Site Wizard) 98, 460, 461
Local pane (Publishing view) 462
local server 65

local site pane 99
Local To Remote arrow (in Publishing view) 446
Local XML files group (Data Source Library) 332
Location dialog box (New Site) 82
Location field
  Connection Settings dialog box 443
  Export dialog box 203
  Import Site Wizard 97, 460
  New Site 81
Lock Aspect Ratio option 215, 218
Log Changes During Publishing option 445
login controls 382
logs, publishing 54
Log tab (Publishing Status) 447
Lost Code Snippets (Code View toolbar) 27
Lucida Calligraphy (Properties panel, Text) 266

M

Macintosh
  fonts 250
  Safari browser, previewing page on 41
main.dwt file 110
Maintain The Site Using Hidden Metadata Files 49, 473
Manage Add-ins dialog box 104
Manage Content Regions button (Master Page toolbar) 141
Manage Content Regions dialog box 141
managed site list 83
Managed Sites List dialog box 472, 477
Managed Sites List (Open Site dialog) 11
Managed Sites List (Site menu) 477
Manage Editable Regions button (DWT toolbar) 305
Manage Editable Regions dialog box
  naming and adding editable region 306
Manage Editable Regions (Format menu) 117
Manage Styles panel 7
  activating 301
  New Style 311
managing Web sites 459–487
  backing up server-based sites 459–465
  editing server-based sites 465–472
  importing and exporting Web Packages 489–493
  using hidden metadata files 14
  using Managed Sites List dialog box 477
  using Search Engine Optimization (SEO) Reports 493–497
  using Site Settings dialog box 472–477
  using subsites 485–489
manifest file for add-ins 101
Manual Style Application 65
margins
  modifying for embedded images 196
  modifying for image in Modify Style dialog 196
  setting for embedded images 194
  setting for <h1> tag 293
  setting for heading (<h1>) tags 293
  setting in New Style dialog, Box category 273
  setting to float container in middle of screen 273
  styling for <body> element 287
Margins and Padding (Visual Aids menu) 34
markup. *See also* HTML
  separation from code in ASP.NET 381
master.dwt file 84
Master Page (New Page dialog box) 384
master pages 383–399
  adding navigation control 400
  converting a DWT to 383–390
  converting HTML image tags to image controls 391–395
  creating new page from 396, 408
  linking new page to navigation menu 397
  pasting menu navigation control into 406
  using as template for attached content pages 136–144
Master Page toolbar
  Manage Content Regions button 141
  Template Region Labels button 142
Match Editable Regions dialog box 126
Maximum Simultaneous Connections 443
Media Elements 7
Media menu 5
  Deep Zoom 223
  Silverlight Video 213
<MediaSource> tags 220
MediaSource tag value 220
media types 322
menu controls 400
  replacing list of navigation links with 403–407
menu division
  linking to default.html file in all folders 309
  styling 275
    adjusting positioning 278
menus 1
  context 12
  menu bars in default workspace 4
Menu Tasks pop-up, Choose Data Source 404, 407
Merge Cells command 336
Message form field 346
messages
  context-sensitive, on status bar 8
  language for server messages 51
metadata
  enabling hidden files to maintain site 49
  file system publishing and 454
  hidden file found by PHP script 369
  Maintain The Site Using Hidden Metadata Files 14
  publishing by third-party FTP client and 449
  selectively using or not using 473
  site 9
meta tags
  adding to DWT <head> section's editable region 307
  DWT, selecting, copying, and pasting into master page 388
  Generator and ProgID tags 54
  placing in ContentPlaceHolder control 141
Microsoft Deep Zoom Composer Forum 228
Microsoft Expression Design
  making default editor 58
Microsoft Expression Design community tutorials 199
Microsoft Expression Design Step by Step 271

Microsoft Expression Encoder online help 222
Microsoft-hosted jQuery library 353
Microsoft .NET Framework versions 436
Microsoft Office Access databases, displaying information from
in GridView control 415–421
in ListView control 422–430
Microsoft Office FrontPage 95
Microsoft Office FrontPage Server Extensions. *See* FrontPage Server Extensions
Microsoft Office Live Small Business 439
Microsoft Official Deep Zoom Composer Blog 227
Microsoft SQL Server databases 415
Microsoft Visual C# 67, 137
Microsoft Visual C++ 101
Microsoft Visual Studio 383
Microsoft Visual Web Developer 383
Microsoft Web Platform initiative 71
MIME types 228
Modify ID (Layers panel) 329
Modify Style dialog box 181
changing footer style 294
Layout, changing float 196
More Colors dialog box 286
changing color for header tags 288
More Swatches button (Properties panel, Appearance) 261
choosing a swatch 263
Move to Center command 20
Mozilla Firefox
comparing IE8 with Firefox 3.6.2 preview 42
downloading and installing 38
installation location 39
mp4 video/mp4 MIME type 228
MSDN Resources for Deep Zoom Composer 227
multimedia display area, creating 324–331
MySQL databases 436
My Web Sites folder 81

N

Name form field 358
valaidation of user input 359
naming files and folders 299
navigation
adding PHP page to 364
designing for a site 297
folder-based 67–70
linking new pages to 113–116
linking page to master page 397–399
setting up for a site 309–311
studying in Hyperlinks view 20
styling navigation menu links for printing 319
styling site navigation menu 311
transferring contents of division tag into ContentPlaceHolder control 395
navigation controls 144, 382, 400–411
adding to page 400–407
creating new page and adding to site map file 408–410
navigation links to folders 50
nested lists 164–166, 167
.net (network site) domain extension 438
.NET Framework versions 436
New dialog box 81
ASP.NET category, Master Page 137
ASP.NET category, Web User Control 145
General category, Dynamic Web Template 121, 302
Programming Language field 137, 146
New Document button (Common toolbar)
ASPX 65
Create From Dynamic Web Template 111
Create From Master Page 142
HTML 67
PHP 74, 151
New Document dialog box
setting Name, Width, Height and Resolution 259
New Editor button (Open With dialog) 58
New Extension button (Application Options dialog) 57, 59
newlines, in code 63
New Page dialog box
ASP.NET category 384
Master Page 384
New Publish Location dialog box 98, 454
importing site to local computer 460
Select 455
New Site Location dialog box 82, 479
New Style (Apply Styles panel) 273
New Style Copy command 183
New Style dialog box
Background category 273
Box category 273
changing color of all heading tags 288
changing float for image 253
Description field 274
Font category 286
Font-Size field 250
Layout category 277
modifying definition list appearance 185
modifying style of ordered list 183
multi-font specification 250–252
Position category 273
positioning content and menu divisions 278
styling a table 178–180
styling controls division 325
styling footer hyperlink 295
styling header 274
styling header division tag 292
styling links 289
styling menu list items 312
styling new element inside #controls division 328
styling site navigation 311
styling unordered list 181
Next Bookmark (Code View toolbar) 27, 29
Next Code Hyperlink (Code View toolbar) 27
Next Result button (Compatibility panel) 239
nodes
selecting from XML file and inserting into page 333
XML SiteMap 409
Normal Paragraphs Without Line Breaks option (Paste Text) 249
numbered lists 163. *See also* lists; ordered lists
creating 167
modifying style 183

O

<object> tag (Silverlight Video) 216
Office Live Small Business 439

Official Team Blog for Expression Encoder 222
 (ordered list) tags 163
  modifying style with CSS 183
"Older" Files Are More Than field (Reports tab, Application Options) 60
On Load settings
  Insert Silverlight Video dialog box 215
  Select A Video dialog box 218
opacity
  fill and stroke 261
Open Attached Dynamic Web Template command 129, 365
Open File dialog box
  opening an image for editing 198
Open (File menu) versus Open Site (Site menu) 10
opening sites 9–13
  viewing all folders and files in a site 12
  workspace with open site in Folders view 11
Open Site dialog box 11, 48
  Add To Managed List 11
  opening server-based site 465
Open Site (Site menu) 10, 48, 74
Open With dialog box 57
  Browse For More 58
  Extension field 59
Open Your Remote Website In Expression Web command 457
Opera 38
Optimize HTML During Publishing check box 52, 445
optimizing HTML during publishing 52–54
Options for Custom Form Handler dialog box 348
Options label (Publishing tab, Site Settings dialog) 476
ordered lists
  creating 167
   tags 163
  styling with CSS 183
.org (organization) domain extension 438
Organizational site templates 80
Orientation field (Layout, Tag Properties) 405
Outer Glow (Effects) 262
Overflow field (Layout category, New Style dialog) 292
Overlay Layout (SuperPreview) 44
Overwrite option (Save Embedded Files dialog) 193

P

padding
  adding to image file 193
  Margins and Padding, Visual Aids menu 34
  setting for <body> element 287
  setting for page elements 281–283
  setting in New Styles dialog, Box category 273
  whitespace as 208
Page Editor Options dialog box 61–65, 236
  Authoring tab 62, 64
  setting default DOCTYPE 237
  tabs 62
Page (New dialog box) 302
Page Properties dialog box
  changing page title 314
  page description and keywords 315
pages
  displaying page titles 19
  opening a page 20–22
  opening without a site 3
  previewing, options for 474
  using page views 22–30
page/site concept 3, 5
page views 22
  Code view 25
  Design view 22
  Split view 23–25
paging
  enabling in GridView control 420
  enabling in ListView control 425
Paintbrush tool 262
  dragging line for footer 263
Palatino font 250
Palatino Linotype font 250
panels 1
  add-ins 101
  applying AutoHide 7, 30
  in default UI screen 4
  in default workspace 4
  ordering within a group 8
  undocking and moving 8
Panels menu 6
  AutoHide All Panels 30
  Reset Workspace Layout command 9, 32
  Snapshot 30–35
  Toolbox 344
paragraph <p> tags
  changing to heading tags 176
  changing to <ul> tag 368
  lists and tables versus 177
Parameter Info (Code View toolbar) 27
parent element, selecting complete content in Design view 114
passive FTP 61, 443
Paste Text dialog box 249
paths
  FTP site import location 460
  Local Copy Location field 461
Path To PHP Executable For Viewing PHP Pages 56
pdb Symbols file MIME type 228
PDF files 202
Pen tool 264
Permissions dialog box 74, 78
Personal site templates 80
  creating site based on 81–83
Phone form field 358
  modifying using Tag Properties panel 360
  setting up 346
  validation of user input 359
Photoshop
  .psd files 202
  using files in a Web page 209–213
Photoshop Maximize Compatibility format option 210
PHP 71
  default file and site navigation 310
  file types included in PHP files 151
  installing and configuring latest thread safe version 71–76
  installing and configuring older version 76–80
  page previews 50
  using Expression Development Server for previews 474
php-cgi.exe file 74
PHP executables 50
  path to, for viewing PHP pages 56
PHP include files 107, 150–156
  attaching DWT to PHP file 150
  creating a new PHP page 150
  creating file to hold content for inclusion 151–153
<?php (phpinfo()) ?> tag 75
PHP tools in Expression Web
  creating automatic image gallery 362–375
  exploring power of generic include files 378
Picture dialog box 188
  selecting an image 135

Picture File Type dialog box 191
Picture Properties dialog box
Appearance tab 189, 208
Size group 193
Wrapping Style and Layout groups 194
Wrapping Style group 208
Browse button beside Picture field 135
Wrapping Style 242
pictures. *See also* images
inserting into a Web page 188
Picture tab, Page Editor Options dialog box 62
player for Silverlight videos 213–222
customizing 218–222
playlist (Silverlight Player) 220
plug-ins (jQuery) 351
linking to plug-in and using it for input validation 355–361
Png files 202
opening with Expression Design 58
position
setting for page elements 281–283
Position category (NewStyle dialog) 277
positioning content and menu divisions 278
setting container position 273
setting header position 275
setting site title position 292
positioning
setting for player control 329
Positioning option (Quick Tag Selector) 25
post and get methods 348
practice files 2, 47, 108, 157, 231, 257, 351, 381, 435
presentational markup, shift to semantic markup 176
presentation, separation from content 247
Preview button (Common toolbar) 36
information on browser sizes 37
Preview In Browser (File menu) 38
Preview in Multiple Browsers 40
previews 50
ASP.NET page, through Expression Development Server 397
browser previews of pages 36–41
browser previews using SuperPreview 41–45
browser view of pages in Snapshot 29–34
dynamic page, with Expression Development Server 65
through FTP connection 470
HTML pages, in Expression Development Server 70
print preview for pages 321
WYSIWYG 33
Preview tab (Site Settings dialog) 49, 474
Use Microsoft Expression Development Server For All Web Pages 313
Preview Using Custom URL for This Website 471
Preview Using Custom URL For This Website 50
Preview Using Website URL 50, 474
Previous Bookmark (Code View toolbar) 27
Previous Code Hyperlink (Code View toolbar) 27
Previous Result button (Compatibility panel) 239
print preview 321
print version of pages, creating style sheet for 316
editing CSS style sheet directly 318
Problems menu (Reports view) 17
Problem Summary in Compatibility panel 240
ProgID tags, removing 54
Programming Language field (New dialog) 137, 146
C# or Visual Basic for ASP.NET pages 146
projection (media type) 322
Properties dialog box 77
for folders 73
Security tab 73
Properties (Expression Design) 199
Effects group 206
Properties panel
adjusting rectangle's appearance 260
Appearance tab, contents of 261
slice names and export format 268
Text group 266
viewing and manipulating slices 200
Proxy Settings button (Application Options dialog) 56
.psd files (Photoshop) 202, 209. *See also* Photoshop
required file extension to import Photoshop files 210
psx Managed Jscript file MIME type 228
public files, folder for 443
Publish All Files (Site menu) 452, 456
Publish Changed Files (Site menu) 451
Publish Changes To The Server Later option 469
publishing
considering hosting requirements 436
considering security requirements 441
defined 435
file system 454–458
logging changes during 54
New Publish Location dialog box 98
optimizing HTML during 52–54
original files, Do Not Publish 213
protocols available in Expression Web 439
Publishing tab, Site Settings dialog box 52
selected files 478–485
setting publishing destinations 15
subsites 488
using FTP 441–448
using HTTP (FrontPage Server Extensions) 449–453
using subsites 485–489
publishing destinations 442
adding, editing, and deleting 52
Publishing Destinations field in Site Settings dialog box 476
Publishing Settings (Site menu) 449
Publishing Status panel 99, 446
file system publishing 456
HTTP (FrontPage Server Extensions) 451
Queued, Failed, Completed, and Log tabs 447
site imported via FTP 463
Publishing tab (Site Settings dialog) 52, 442
contents of 476
Include Subsites 488
Publishing Destinations group 445
Publishing view 14, 445–448
file system publishing 456

Publishing view *(continued)*
HTTP through FrontPage Server Extensions 451
importing site in 462
Local Site and Remote Site panes 446
Publishing Status panel 446
SelectivePublish, Connect To drop-down 480
Publish Selected Files option 484
py IronPython file MIME type 228

Q

quality settings for images 191
exported from Expression Design 202
Queued tab (Publishing Status) 447
Quick Tag Editor 161
changing <p> tag to <ul> tag 368
Quick Tag Selector 113, 117, 161
displaying options 25
using creative cut and paste with 389

R

Raster format, images exported from Expression Design 202
Recalculate Hyperlinks 269
Recalculate Hyperlinks (Tools menu) 203
"Recent" Files Are Less Than field (Reports View tab, Application Options) 60
Recent Sites (Site menu) 464
rectangle, creating as Web page template container 259
Rectangle tool 259, 266
Reformat CSS command 319
Reformat HTML command 276
Reformat XML command 404
Refresh Changed Results button (Compatibility panel) 239
Refresh Contents (ASCX file) 149
Refresh (View menu) 208
Region Name field (Editable Regions dialog) 118
Regions box (DWT toolbar) 305
registered trademark symbol 103
registering domain names 437
using Microsoft Office Live Small Business 439
registrars (domain name) 438
Registration Point button (action bar) 201
registrations (domain name), reselling 438
Remote pane (Publishing view) 462
Remote Site Editing Options dialog box 469
remote site pane 99
Remote view (SuperPreview) 41
Remote Web site view, working with hosting service default files 448
Remove Comments (HTML optimization) 53
Remove Generated HTML (HTML optimization) 54
Remove Tag command 193
Remove Tag option (Quick Tag Selector) 25
Remove Whitespace (HTML optimization) 53
renaming image files 193
Replace Files dialog box 207
report panels, functions of 239
reports
Reports View tab (Application Options) 59
Search Engine Optimization (SEO) Reports 493–497
Reports view 14–17
All Files 17
Site Summary 16
Resample menu 190
Resample Picture To Match Size 191
reseller accounts (hosting) 437
reselling domain name registrations 438
Reset Workspace Layout (Panels menu) 9, 32
Resize Options menu 193
resizing images 193
root folder 12
creating 298
saving default.master file in 385
saving file created from master page into 396
rotation, changing for embedded image 205
rows, inserting into tables 160
RSS feeds 6
importing feed.xml example file 332
Ruler and Grid tab (Page Editor Options dialog) 63
Run Compatibility Checker 238
Run Compatibility Checker button (Compatibility panel) 239
run-time includes 154
run-time template functionality 108

S

Safari 38
previewing pages with SuperPreview 41
sans-serif fonts 319
Save All (Common toolbar) 366
Save All (File menu) 302
Save As dialog box 66
Change Title button 308
saving CSS style sheet in folder 300
saving file created from master page 396
saving master page as default.master 385
saving PHP auto gallery file 371
saving PHP file into root folder 364
saving template design 269
Save Embedded Files dialog box
Action, Overwrite 193
and image file exported from Expression Design 204
Main.css file, overwriting and saving 313
Picture File Type 191
saving files, open pages attached to DWT 116
screen (media type) 322
script comments 53
scripting functionality 323–350
client-side functionality
creating and using HTML forms 343–349
customizing Data View source code 337–343
using Data View in HTML page 332–337
using Layers and Behaviors 324–331
server-side scripting 323
server-side vs. client-side scripting 323
ScriptManager controls 431
Seadragon Ajax for Deep Zoom image 223
Search Engine Optimization panel 494
depth and detail of information in 496
Search Engine Optimization (SEO) Reports 493–497

search engine results
domain names and 437
heading (<h1>) tags and 293
improved, with folder-based navigation 310
keywords and 315
page description and keywords 315
Search Engine Results Page position (SERP) 493
secure sessions (https) 441
Secure Shell (SSH), FTP over 439
Secure Sockets Layer (SSL), FTP over 439
Secure Sockets Layer (SSL/HTTPS) 449
security requirements for publishing 441
Security tab (Properties dialog) 73
Select All button (toolbar) 463
Select A Master Page dialog box 142, 396, 408
Select An Include File dialog box 153
Select A Video dialog box 214, 218
Select Block (Code View toolbar) 27
Selected Slice Properties pane (Export dialog) 202
Select File To Include dialog box 372
Select Image dialog box 393
selection tool (Expression Design) 200
selectively publishing specific files 478–485
Select Photoshop (.psd) file dialog box 210
Select Style Sheet dialog box 86, 88, 325
Select Tag button (Code View toolbar) 27
Select Tag Contents option (Quick Tag Selector) 25
Select Tag option (Quick Tag Selector) 25
semantic markup 173–176
header tags 176
shift from presentational markup 176
 and (emphasized tags) 173–175
Send To Other option (Form Properties dialog) 347
SEO Checker dialog box 494–496
All Pages option 495
SEO Reports. *See* Search Engine Optimization Reports
Serif font family 250
serif fonts 319
server-based sites
backing up to local computer 460–465
editing 465–472
opening via FTP or HTTP (FrontPage Server Extensions) 469
Server Message Language setting 51, 475
servers
choosing server to host your site 436
encryption of data in secure sessions 441
Expression Development server as local server 65–71
server-side includes 108, 154
content from PHP auto gallery file 372
server-side run-time versus save-time templates 397
server-side scripting
ASP.NET 381
ASP.NET Development Server and 474
versus client-side scripting 323
Set Page Title dialog box 121, 150
changing template 308
Set Text Of Layer dialog box 330
for Windows Media interactive button 331
Settings dialog box
Preview tab
Use Microsoft Expression Development Server 313
Settings group (Connection Settings dialog) 443
Use Encrypted Connection option 450
SFTP 439
shading, setting for media player 329
Show Data (Data Source Library panel) 333
Show Page Titles context menu 19
ShowPosterFrame option 218
Show Set Properties button (Tag Properties) 393
Show Status Bar option (Application Options dialog) 55
Show Template Region Labels button (DWT toolbar) 305
ShowVideoPaused and ShowVideoPlaying options 215
Silverlight applications
Deep Zoom technology 222–227
MIME types 228
Silverlight 3 Canvas image format 202
Silverlight 4 / WPF Canvas image format 202
Silverlight.js file 219
video, using in Web page 213–222
silverlightgallery.xap file 219
simultaneous connections, maximum number 443
Single Item View 333
site architecture and navigation, designing 297–316
adding pages based on DWT 307
adding titles to pages 314
attaching style sheet to DWT 305
creating and linking external CSS file 300–302
creating DWT 302–304
creating unordered list of folders for navigation menu 298
navigation menu 309–311
styling navigation menu 311–313
site management 459
backing up server-based sites 459–465
editing server-based sites 465–472
importing and exporting Web Packages 489–493
using SEO Reports 493–497
using subsites 485–489
site map files 403–407
adding new page 408–410
configuring as data source 404
SiteMapPath controls 400
Site menu 6
Export To Web Package 489
Import 96, 460, 492
Managed Sites List 477
Manage Sites List 11
New, One Page Site 258
New Site 81
Open Site 48, 74, 465
Open Site command 10
Publish All Files 456, 482
Publish Changed Files 451, 483
Publishing 445
Publishing Settings 449
Recent Sites 464
Site Settings 14, 442
sites
closing and opening 3
creating empty site and importing files and folders 91–96

sites *(continued)*
creating from Expression Web template 80–91
creating using site template 80–98
importing with Import Site Wizard 96–101
opening a page 20–22
opening a site 9–13
page/site concept 3, 5
previous open site, opening automatically 10
using site views 14–21
Site Settings dialog box 14, 48–54, 472
Advanced tab 51, 475
General tab 49
making global changes and troubleshooting 473–477
PHP executable for previews 78
previewing site using custom URL 471
Preview tab 50, 70, 474
Publishing tab 52, 442, 476
setting custom preview URL 471
Site Settings (Site menu) 14, 48
Site Summary report 16
site views 14–20
Folders 14
Hyperlinks 18–20
Publishing 15
Reports 15–17
size
changing for images 193
images to import from Photoshop 210
Resample Picture To Match Size 191
Resize Picture To Match Size 193
settings for pictures 190
slices
contents and aspects of slice in Properties panel 200
creating and exporting for graphics template 268–270
editing images in Expression Design 199–201
exporting from Expression Design 201
Slice tool 200, 268
"Slow Pages" Take At Least field (Reports View tab, Application Options) 60
Small Business site templates 80
Snapshot panel 29–34
rendering code for Internet Explorer 7 32
sorting, enabling in GridView control 420
 tags, styling with CSS 290–292
Specify Size (Picture Properties dialog) 190
Split view 23–25, 75, 83, 109
HTML instruction in 113
SQL Server databases 415, 436
SSL certificates 439, 441
SSL (Secure Sockets Layer)
FTPS publishing 439
HTTPS publishing 450
standard controls 382
Standard toolbar 1
Include Page button 132, 133, 225
opening in workspace 132
Publish All files and Publish Changed Files 452
starting Expression Web 4
without a site open 10
Startup options (Application Options dialog) 55
states of links 289
StaticDisplayLevel field (Behavior group, Tag Properties) 405
static pages (.htm/.html) 436
status bar 8
in default UI screen 4
Show Status Bar option 55
Stroke button (Properties panel, Appearance) 261
choosing stroke 265
More Color Swatches 264
putting black outline around text 266
Stroke Gallery (Properties panel, Appearance) 261
Basic Stroke 262
Design Elements, Geometric Flowers with Stars 265
General group, Thin Multi-layer 263
Ink category, Fuzzy Brush 264
Watercolor category, Fuzzy Thick Wet Brush 264
Stroke Opacity (Properties panel, Appearance) 261
Stroke Width (Properties panel, Appearance) 261
 tags 173–175
Style Application Mode 8
Style Application toolbar 65
Style Builder dialog box 180
Style menu
Defined Term (<dt>) 169
Definition 171
styles. *See also* CSS
CSS (Cascading Style Sheets) 63
Manage Styles Panel 7
style sheet links in head sections of pages 123, 140
style sheets. *See also* CSS
for add-ins 101
attaching to DWT 122
changing 84–91
styling content presentation 177–187
CSS (Cascading Style Sheets) 187
definition list 185
ordered list style 183
table <th> elements 177–180
unordered list 181–185
styling for alternative media 316–322
styling major HTML elements 284–296
creating CSS styles for HTML markup 284
<body> tag 286–288
changing color of all heading tags 288–290
footer division of page 294
header division of page 290–294
subsites 485–489
converting folders to 486
converting to folders 488
including when publishing 52
publishing 488
SuperPreview 41–45
accessing functions of 43
Baseline browser and Comparison browser panes 42
checking any page on the Internet 45
DOM Tree View 44
opening independently of Expression Web 45
Overlay Layout 44
Remote view 41
Vertical Split Layout 44
symbols, inserting 102
Synchronize Files button 463

T

Tab key, inserting table rows 160
Table menu 6
Insert Table 159

tables
setting up table for Data View 334–336
styling with CSS 177–180
using properly 158–163
tab order for form fields 345
email field 346
tag case, in Code view 63
Tag Properties option (Quick Tag Selector) 25
Tag Properties panel 8, 359
Interval field, Timer control 431
Layout group 393
modifying form field tag properties 360
modifying image control 392
Set Properties on Top button 361
Show Set Properties button 393
Triggers field 432
use in modifying ASP.NET controls 383
working with site navigation control 405
Target Frame and Parameters settings for hyperlinks, 335
Tasks button (AdRotator) 412
<td> and <th> tags
editing 161–163
styling <th> tags 177–180
templates 80–91
adding a DWT to a PHP file 363
benefits of using 107
changing style sheet linked to DWT 85–90
converting a DWT to a master page 383–399
creating site with 80–85
Data View 343
different definitions of 107
formats 95
linking jQuery library to 353
Silverlight 214, 219
Template field, Select A Video dialog box 218
.tpl file extension used by PHP scripts 58
understanding concepts 108
using ASP.NET master pages 136–145
using ASP.NET web user controls 145–150
using Dynamic Web Templates (DWTs) 109–131
using Include Page feature 128–137
using PHP include files 150–156
working with graphics template 257–271
temporary files 51
Temporary Files label (Advanced tab, Site Settings dialog) 475
text 157
color set in <body> tag 288
styling with cascading style sheets 247–256
Text-Align field (New Style dialog) 312
TextArea Box Properties dialog box 346
text area field, inserting into a form 346
Text Box Properties dialog box 345
Text-Decoration field (New Style dialog) 290
removing underline from link 312
text fields, inserting into a form 345
text files 285
Text group (Properties panel) 266
Text-Indent field (Block category (New Style dialog) 292
Text To Display field (Edit Hyperlink dialog) 335
Text tool 266
Text-Transform field (New Style dialog) 185
Thin Multi-layer strokes 263
thumbnails
AutoThumbnail feature 63
representing Silverlight videos 220
TIFF files, image export 202
time increments for building reports 60
Timer controls 431
trigger 432
titles
adding to pages in a site 314
Set Page Title dialog box 308
setting for DWT master page 121
setting for PHP pages 150
Show Page Titles command 19
styling site title 292
Toggle Bookmark button (Code View toolbar) 27
inserting a bookmark 28
toolbars 1
Common toolbar 6
currently active 6
in default workspace 4
SuperPreview 42
Toolbars menu 6
Code View 27–29
Dynamic Web Template 116, 305
Master Page 141
Standard 132
Style Application 65
Toolbox (Expression Design) 199
Selection button 200
slice tool 200
Toolbox panel 7
AJAX controls 431
ASP.NET controls 382
ASP.NET controls, Data group 423
ASP.NET group 381
autohiding 301
data controls 415
Form Controls 344
Image, in ASP.NET Controls Standard group 392
Input (Reset) and Input (Submit) 346
Input (Text) 345, 346
inserting ContentPlaceHolder control 389
Manage Styles tab 7
Paintbrush tool 262, 263
Rectangle tool 259
Selection tool 263
Slice tool 268
Standard group, AdRotator control 411
Text Area 346
Tools menu 6
Accessibility Reports 244, 246
Add-in 103
Application Options 55, 74
Page Editor Options 62, 237
Recalculate Hyperlinks 203, 269
SEO Reports 494
tooltips 6
displaying for status bar items 8
.tpl file extension 58
transfer as ASCII 61
Tracing Images, removing attributes 54
transformations, XML documents 339
TreeView controls 400
trigger for timer control 432
Triggers field (Tag Properties) 432
tty (media type) 322
tutorials, Expression Design 199
tv (media type) 322

U

 (unordered list) tags 163
changing <p> tag to 368
within <li> tags 165

 (unordered list) tags *(continued)*
styling 181
underlining for links 312
underscore (_), files and folders beginning with 475
Unicode (UTF-8) encoding 475
Uniform Resource Locators. *See* URLs
unlinked files in a site 16
grouped by item type 17
unordered lists
creating 164–166
creating within ordered lists 164
styling site navigation menu list 311
styling with CSS 181–183
 tags 163
using for navigation menu 298
using PHP to create list of files in a folder 367–369
unstyled HTML elements, causing cross-browser differences 287
Untitled_1.htm page 10
up and down arrows on keyboard, cycling through Silverlight templates 214
Update Attached Pages button (DWT toolbar) 305
Update Files dialog box 88, 89
Update From Source command 211
UpdatePanel (AJAX group in Toolbox) 432
pasting AdRotator control into 433
UpdatePanelTrigger Collection Editor 432
uppercase, changing definition list text to 185
URLs
custom preview URL 470
domain name as base 437
previews using Web site URL 474
publishing into 436
Web site or custom, for previews 50
Use A PHP Executable For Only This Website (Site Settings dialog) 50
Use Encrypted Connection option 450
Use Microsoft Expression Development Server (for previews) 50, 70, 474
Use Passive FTP
Application Options dialog box, FTP tab 61
Connection Settings dialog box 443
user input
sending to results page via contact form 343–349
validating using jQuery plug-in 355–361
user interface (UI) 1–46
browser previews 36–41
default screen layouts 2–9
editing window 4
menus, toolbars, and panels 1
opening a page 20–22
opening a site 9–13
page views 22–30
site views 14–21
Snapshot panel 29–34
SuperPreview 41–45
visual aids 33–36
user name and password for publishing 436
Credentials field, Connection Settings dialog 443
HTTP (FrontPage Server Extensions) publishing 449
HTTP, using FrontPage Server Extensions 441
Use The PHP Executable Set In The Application Options (Site Setttings dialog) 50
U.S. Rehabilitation Act 244

V

validation
ASP.NET controls for 382
using jQuery plug-in to validate form input 356–361
validation engine (W3C) 243
validation service 64
validity 231–244
DOCTYPE, understanding and changing 232–237
verifying W3C validity 238–243
Variable Bit Rate (VBR) 215, 218
variables (PHP) 370
hard-coded, replacing with variables set in page 373–377
vector drawing tool (Expression Design) 199
Verify Well Formed XML command 404
version number, adding to default installation folder location 72
Vertical Margin (Picture Properties) 194
Vertical Split Layout (SuperPreview) 44
videos
MIME types 228
object tag text, accessibility considerations 244
using Silverlight video in Web page 213–222
View Log File button (Publishing tab, Site Settings dialog) 445
View menu 5
Actual Size 205
Formatting Marks 35
Refresh 203
Show command, Grid, Guides, and Edges 259
Toolbars 6
Code View toolbar 27
Master Page 141
Standard 132, 226
Visual Aids 34
Block Selection and Visual Borders 158
views 1
browser view of Web page using Snapshot Panel 29–33
page views 22–30
Code, Design, or Split 23
Code view 25–29
Design view as default 22
relationship between Design and Code views 24
site views 14–21
Folders, Publishing, Reports, and Hyperlinks 14
Hyperlinks 18–20
Publishing 15
Reports 15–17
using visual aids in Design view 33–36
Visible Borders 158
visited state (links) 289
visual aids 8, 33–36
Block Selection label 24
formatting marks 35
Margins and Padding 34
showing 259
turning on and off 34
using to view ASP.NET controls 382
Visual Aids button on Status Bar 33
Visual Aids button on status bar 33
Visual Aids menu 33–36, 158
Visual Basic 137
Visual Studio 383
Visual Web Developer 383

W

W3C (World Wide Web Consortium)
  standards for HTML/XHTML validity 231
  list of DOCTYPEs 233
  validation engine, testing page on 243
  verifying and addressing validity standards 238–244
Watercolor category, Fuzzy Thick Wet Brush (Stroke Gallery) 264
WCAG Web Content Accessibility Guidelines 244
WDP (HD Photo) files 202
WebDAV 440
Web Name field (General tab, Site Settings dialog) 49, 473
Web Package Properties dialog box 491
Web Packages
  exporting 489–492
  importing 492
WebParts controls 382
Web.sitemap file 400
Web standards, key principles 231
Web user controls 108. *See also* ASP.NET; controls
  adding 145–150
  control displaying variables of server 66
whitespace
  in code 63
  removing from HTML code 53
  removing from HTML during publishing 53
Width and Height settings
  Deep Zoom Composer images 223
  images exported from Expression Design 202
  images in Expression Design 201
Width field (Layout, Tag Properties)
  GridView control 421
Window menu 6
  Close All Pages 21
Windows
  network and Internet connections 56
  readers running Windows XP or Vista 2, 47, 108, 158, 232, 351, 381, 436
  servers running or not running 436
Windows Installer 71
Windows Media interactive button 331
Windows Start menu
  All Programs, Microsoft Expression 259
  opening SuperPreview 45
Windows Web Application Gallery 71
Word Wrap (Code View) 27
workspace
  default layout 4–9, 41
  with open site in Folders view 11
  Reset Workspace Layout (Panels menu) 32
World Wide Web Consortium. *See* W3C
World Wide Web Consortium–valid (W3C-valid) HTML code 83
WPF Drawing Brush image format 202
Wrapping Style (Picture Properties) 194, 208, 242
Wrap Tag option (Quick Tag Selector) 25
WYSIWYG presentation in Design view 33

X

XAML format
  images exported from Expression Design 202
  output containers 203
xaml text/xml MIME type 228
X and Y coordinates of objects
  display by graphics programs 201
xap application/x-silverlight-app MIME type 228
XHTML 1.0 Transitional DOCTYPE 233, 236
XHTML files
  checking compatibility 240
  setting default DOCTYPE 236
  understanding and changing DOCTYPEs 232–237
  verifying and addressing accessibility 244–247
  verifying and addressing W3C validity 238–243
XML (Extensible Markup Language)
  manifest file for add-ins 101
XML files. *See also* data controls; Data View
  data source for AdRotator control 412–415
  data source for Data View 332
  displaying data from 6
  HTML tags in 341
  site map 404–411
  transforming 339
  Web.sitemap 400
  writing dynamically into HTML pages 343
xpi application/x-xpinstall MIME type 228
XSL files
  Disable-Output-Escaping property 341
  editing feed.xsl file 339
  editing to correct rendering of HTML tags 339–341
  feed.xsl and xslview.js 334
  formatting and functionality for Data View 334
XSLT (Extensible Stylesheet Language Transformations) 339

About the Author

Chris Leeds is a long-time digital photographer and Web enthusiast who has been a Microsoft Most Valuable Professional (MVP) for Microsoft Office FrontPage and Expression Web for eight years. He is also a Network Partner with Microsoft's Website Spark program (*www.websitespark.com*), commentator at Lockergnome (*www.lockergnome.com*), and a software reviewer on Bright Hub (*www.brighthub.com*).

Chris developed a software product called ContentSeed (*www.contentseed.com*) with which users can create Web pages that they can edit and manage using only a browser.

Chris was a technical reviewer for *FrontPage 2003: The Missing Manual* (O'Reilly), the author of *Microsoft® Expression® Web Step by Step* and *Microsoft® Expression® Web 2 Step by Step* (the previous versions of this book), and co-author for the upcoming book *Microsoft® Expression® Blend 4 Step by Step* (Microsoft Press). He has developed several tutorials about FrontPage, and he hopes to continue helping the user community through the Web site *www.expressionwebstepbystep.com*, from which he'll try to answer questions regarding this book and Microsoft Expression products.

What do you think of this book?

We want to hear from you!

To participate in a brief online survey, please visit:

microsoft.com/learning/booksurvey

Microsoft® Press

Stay in touch!

Microsoft Press® Book Connection Newsletter

microsoft.com/learning/books/newsletter