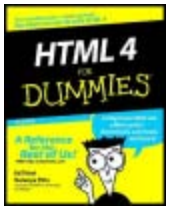

HTML 4 for Dummies, 4th Edition

by Ed Tittel and Natanya Pitts

ISBN:0764519956

John Wiley & Sons © 2003 (408 pages)

Whether your goal is to build a simple, text-oriented Web site or one loaded with frames, graphics, and animation, this step-by-step book will put you on the right track.

Table of Contents

HTML 4 For Dummies, 4th Edition
Introduction
Part I - Meeting HTML in Its Natural Environment
Chapter 1 - The Least You Need to Know about HTML and the Web
Chapter 2 - HTML at Work on the Web
Chapter 3 - Creating Your First HTML Page
Part II - Getting Started with HTML
Chapter 4 - Structuring Your HTML Documents
Chapter 5 - Linking to Online Resources
Chapter 6 - Finding and Using Images
Chapter 7 - Top Off Your Page with Formatting
Part III - Taking HTML to the Next Level
Chapter 8 - HTML Tables
Chapter 9 - HTML Frames
Chapter 10 - HTML Forms
Part IV - Extending HTML with Other Technologies
Chapter 11 - Getting Stylish with CSS
Chapter 12 - HTML and Scripting
Chapter 13 - Making Multimedia Magic
Chapter 14 - Integrating a Database into Your HTML
Chapter 15 - How HTML Relates to Other Markup Languages
Part V - From Web Page to Web Site
Chapter 16 - Creating an HTML Toolbox
Chapter 17 - Setting Up Your Online Presence
Chapter 18 - Creating a Great User Interface
Part VI - The Part of Tens
Chapter 19 - Ten Ways to Exterminate Web Bugs
Chapter 20 - Ten HTML Do's and Don'ts
Part VII - Appendixes
Appendix A - HTML 4 Tags
Appendix B - HTML Character Codes
Appendix C - Glossary
Index
Cheat Sheet - HTML 4 For Dummies, 4th Edition
List of Figures
List of Tables
List of Listings
List of Sidebars

Establishing Solid Navigation

The navigation you use on your site can make or break it. If visitors can't find what they're looking for on your site, chances are they will leave and never come back. The type of navigation you use on your site depends on how many pages you put on it — and on how you organize them. If you only have a few pages, your navigation may be a simple collection of links on the home page that help users jump to each page. However, if your site has many pages organized into different sections, your home page may only link to those sections and not to each page.

For example, the `Dummies.com` site houses a large collection of pages organized as a variety of sections; it would be impractical to link to all the pages in any navigation scheme. Also, the site includes articles on a wide variety of topics, as well as book information. The site could be organized into books and articles, but visitors are likelier to be looking for information on a particular subject, so the site is organized by topic. The home page, shown in Figure 18-4, prominently displays these different topic areas on the left.

Figure 18-4: The Dummies.com site is organized by topic.

When you click into one of these topic areas, the remaining topic areas are available in a navigation bar across the top of the page (as shown in Figure 18-5). You don't have to return to the home page to jump from topic to topic.

Figure 18-5: The main topic areas on the Dummies.com site are accessible from the top navigation

bar.

As you can also see from Figure 18-5, each topic has its own sub-navigation area (at left, echoing the layout the home page) that lists sub-topics within the topic. Although the links are different, the general navigation scheme is consistent throughout the site. Thus visitors know what to expect as they navigate around the site.

Finally, the topmost navigation area of each page includes a regular collection of links that appear on every page of the site to help visitors quickly access important areas from anywhere: a site search, help, account information, and a shopping cart. Every page has the same set of links to information about Dummies, the form to register for eTips, how to contact Dummies, the site copyright statement, and the site privacy policy. Like the shopping cart and help links, these links have to be on every page, but need not be displayed as prominently. Adding them to a consistent site footer keeps them accessible to visitors without obscuring key content for any given topic or sub-topic.

If you create a map to aid site development, you can also use it as a guide for the kind of navigational tools you need to create for your site. Consider each page on the map in turn; list the links that each page must include. A pattern normally emerges that can help you identify the main navigation tools your site needs (such as links to all main topic areas and copyright information, as on the Dummies site), as well as sub-navigation tools (such as links to sub-topics on the topic pages).

After you know what tools you need, you can begin to design the visual scheme for your UI. Do you want to use buttons across the top, buttons down the side, or both? Do you need a footer that links to copyright or privacy information? If you have sections within sections within sections, how can you best help people navigate through them? Answering questions like these is the route to a solid navigation system that helps users find their way around your site — letting them focus on what they came for rather than on how to get there.

Whatever navigation scheme you finally devise, always give your visitors a way to get back to your home page from wherever they are on the site. Your site's home page is the gateway to the rest of the site, and if visitors get lost or want to start again, make sure they can get back to Square One with no trouble.

After you design your site navigation scheme and put together a few pages, ask someone who isn't familiar with your site to review it and try to use it. To help them with their testing, give them a list of three or four tasks you'd like them to complete — pages to visit or a form to fill out, for example. If your test visitor gets lost or has lots of questions about how to navigate the site, you may need to revisit your scheme. Your reviewer may also have suggestions on ways to make the navigation features clearer and easier to use. Remember, you know your site and its content like the back of your hand — in effect, you know too much. You may not see gaps in the navigation system that a first-time user will probably turn up right away. It's better to find those gaps during site development than to be pelted with complaints after the site goes live.

Good Linking Practices

The Web wouldn't be the Web without hyperlinks - after all, they can connect your site to the rest of the Web and turn a collection of pages into a cohesive site. But overusing or misusing links can detract from your site and even lose you some business.

Choose your off-site links wisely

Internal linking is almost a walk in the park compared to external linking - after all, when you link to pages on your own site, the pages those links point to are *under your control*. You know what's on them today and what will be on them tomorrow, and even whether they will exist tomorrow. When you link to resources on someone else's site, however, all bets are off. You don't maintain those pages, you can't modify their content , and you certainly won't know when they disappear. Neither will your visitors - till they slam into a `404 Page not found` message (the usual sign of a *brokenlink* that now goes nowhere).

Links to other sites are more useful when they're relatively stable and have less chance of breaking. Consider these suggestions:

- Link to a particular section of a site rather than a specific page on the site. Pages come and go on sites, especially large ones, but the general organization typically stays the same, making sections a better linking bet.
- Carefully choose what sites you link to. Sites maintained by companies don't usually go away, but those maintained by individuals do so more often.
- Rather than linking directly to PDFs or images or other media files on a site, link to the pages on the site that link to those resources. Sites often update the resources and give them new names. The page that links to the resource will almost always be updated to reflect new names, so the page is a safer linking bet.

Good linking practices are not a substitute for regular site maintenance. You should regularly check (once a week if possible, once a month at least) all links from your site to external sites to be sure they're still working. A good Web editor will do the checking for you, as will other site utilities as described inChapter 16.

Even though you don't control the content on the Web sites you link to, providing links to them *implies your support or endorsement of those sites*. When a visitor follows a link from your site to someone else's site, he or she thinks you approve of the content on that new site. That human quirk makes a couple of guidelines necessary:

- **If you don't want to be associated with content on a particular external site, don't link to the site.** Seems obvious, doesn't it? But the only way to find out whether you approve of a seemingly relevant site is to visit it and check it out *before* you link.
- **Periodically review your links to be sure the sites' owners are still the same - and that the content is still appropriate.** When domain names expire, new owners may take them over and create new content that's completely irrelevant - or, worse, damaging to your image (think pornography).

Craft useful link text

The text you associate with links is just as important as the links you choose to use on your site. The text gives users a hint about where the link is taking them so they can decide whether to go along for the ride. For example, `Visit Dummies.com to read more about this book` is much more helpful than `Read more about this book`.

The first bit of text tells visitors they're going to leave the current site to visit `Dummies.com` and read more

about a book there. The second just tells them they're going to read more about the book - and they might be surprised to find themselves flung off one site and onto another.

Generally, when you create link text, let users know the following:

- Whether they're leaving your site.
- What kind of information the page they're linking to contains.
- How the linked site relates to the current content or page on your site

The goal of your link text should be to inform users and build their trust. If your link text doesn't give them solid clues about what to expect from your links, they simply won't trust your links - and won't follow them.

Always avoid the use of 'click here' in any link you create. If you link text is well crafted, you don't need the extra words to prompt the user to click on a link. The link text should speak for itself.

Choosing the Right Bells and Whistles

Media can offer a huge boon for your Web site, adding interactivity and some pizzazz to text and static graphics. However, you should only use media that make your site work better (and/or support the information on it). As a general rule, gratuitously slathering a site with media is bad media. When you choose media for your site, ask yourself the following questions:

- **Why are you adding these particular media to your site?** For example, an audio clip of an interview may be less effective than a transcript. An interactive Flash presentation can demonstrate how a product works or teach someone how to do something - but Flash-enabled content can take longer to download. If a medium doesn't have a clear purpose, you may want to rethink its integration into your site.

- **What are some alternatives to the chosen media?** Not everyone is equipped to view (or listen to) all media formats. Therefore make sure the information offered in your media is also available in an alternative format. For example, if visitors to your site don't have Flash and can't interact with your Flash navigation, how do they get around your site?

- **How large is the media file?** Large files may take a long time to download, so make sure the download is well worth the wait for users. When you do choose to include large files, be sure to let users know *how* large so they can choose whether to download. Also see whether you can split large files into smaller files that visitors can access one at a time.

- **Will visitors need a special plug-in or application to access the media file?** If users aren't set up to play a particular media format, they may skip using the media (or drum their fingers while the application downloads). Be sure you let visitors know exactly what formats your media files are saved in - and the tools needed to view those files - so they can decide whether they want to download the media. (The wise Web-site designer also includes a way for users to bypass the media, which helps to ensure the widest possible audience for the content. More about that in a moment.)

Working with media on the Web is a significant topic in itself. The Resources section at the end of this chapter includes information on Web sites and books that cover media in more detail. Chapter 13 is a good starting point for using media in your Web pages.

Making Your Site Accessible to Everyone

If you optimize your Web site for users with fast Internet connections, large monitors, the latest browsers, and every plug-in known to humanity, you'll be severely limiting the audience for your site. Instead, you should design your site to accommodate as many users as possible with the widest variety of equipment.

The short list of things you can do to make your site accessible includes:

- Providing alternative text for all images and alternative content formats for all media.
- Using Cascading Style Sheets where possible for styling (instead of `<font>` and other markup) when you apply formatting to your documents.
- Reviewing your pages in as many browsers as possible, including a text-only browser so you can get an idea of how the various browsers handle your combination of text and markup.
- Following Web Accessibility Guidelines wherever possible, as outlined by the W3C's Web Content Accessibility Guidelines at www.w3.org/TR/WCAG.

Bobby is a Web-based tool that evaluates your Web page and lets you know exactly how accessible it really is. We strongly suggest that you check all your pages with Bobby to find any accessibility problems you might have overlooked. To run a Bobby scan, visit

http://bobby.watchfire.com/

and type the URL of the page you want to check in the URL: box.

Some Excellent User Interfaces

Spend a few minutes browsing the Web and you'll be bombarded with excellent (and not-so-excellent) examples of user interfaces. How do you know whether an interface *is* excellent? Simple: If you can find what you need on a site with relatively little trouble, and you come away feeling like you had a good experience, you've used a good interface. Two examples of excellent user interfaces are the Amazon.com site and (believe it or not) the IRS site.

Amazon.com

Earlier on, the chapter looked at how a site's goals directly affect its user interface, with an online store as the prime example of defining a site's goals and then designing an appropriate UI. Well, behold a sterling example: Everything about Amazon.com is designed to facilitate the buying experience. This site is divided into sections, each represented by a different color tab across the top. Visit `www.amazon.com` to see what we're referring to.

Regardless of where you are in the site, you can get to any other section — you can always search, access your shopping cart and account, get help, and (of course) check out to buy. The site's system for checking out is broken into easy-to-manage screens, so purchasing isn't intimidating.

Every product page follows the same basic format, whether you're shopping for a book or for a garden hose, large amounts of help and FAQ text are well divided and easy to navigate. Simply click on the HELP link in the upper-right corner.

There isn't anything on the Amazon.com site that doesn't have to do with selling products online. You won't find any gratuitous media, but you will find a special feature that enables you to preview book contents right on the page. The goals of the site are well reflected in its organization and design.

The IRS

The IRS Web site contains a great deal of important information relevant to everyone in the United States. That means the site has to be easily navigable and well organized so users can find what they're looking for. The site is divided into sections, each of which is linked from a persistent navigation bar at the top of the UI, as shown in Figure 18-6.

Figure 18-6: Main areas on the IRS site are accessible from a persistent navigation bar.

Each main area of the site uses the same layout, including a left column of fields for searching both the site in general and specific IRS forms. Then there's a list of contents for the site, links to relevant

resources, and access to topics of discussion. Although the actual content, resources, and topics change from section to section, they are accessible on all pages from the same spot, as shown in Figure 18-7.

Figure 18-7: Each area of the site uses the same general layout.

The inclusion of a persistent link enables a visitor to search for a publication from any page. Such features contribute significantly to one of the site's primary goals: making publications easily available. If you're doing research on a particular tax topic and realize you need a particular publication, you can search for it right from the page that let you know you needed it in the first place. You don't have to jump to a search page first, which saves steps and makes publications that much more accessible.

In general, the IRS site is a great example of making information that users will need most (like publications) immediately accessible. The information on the site is well organized — and it's obvious the site designers work diligently to create a UI that supports users in their search for tax information.

More Resources on UI Design

We've barely scratched the surface of Web and UI design in this chapter, but we're confident it's enough to get started. We also recommend the following Web sites and books on site and interface design if you want to dig into creating great UIs:

- For a crash course on Web design basics read Design Basics from Webmonkey at

  http://hotwired.lycos.com/webmonkey/html/97/05/index2a.html

  In addition, its Site Redesign Tutorial offers an interesting perspective on what it takes to rework a site's des Read it at

  http://hotwired.lycos.com/webmonkey/design/site_building/tutorials/ tutorial4

- Jakob Nielsen is committed to creating accessible Web content. His Web site, http://useit.com, is ch full of resources and articles on creating accessible sites.
- Hey, negative examples are useful too. Web Pages That Suck helps you learn about good design by evalu bad design. Be sure your site doesn't look like any of those featured at www.webpagesthatsuck.com/.
- *Web Design For Dummies* by Lisa Lopuck (Wiley Publishing, Inc.) is another step in the direction of a sophisticated Web site with a knockout look.
- *Web Usability For Dummies* by Richard Mander and Bud Smith (Wiley Publishing, Inc.) can help you fine-tu your site to make it amazingly easy to use (which is a great help in keeping your visitors coming back for m

Part VI: The Part of Tens

In This Part:

Chapter 19: Ten Ways to Exterminate Web Bugs

Chapter 20: Ten HTML Do's and Don'ts

In this part . . .

Here we help you catch potential bugs and errors in your Web pages, cover top do's and don'ts for HTML markup, and point out key HTML resources you can use to extend your knowledge, skills, and sources of inspiration. Enjoy!

Chapter 19: Ten Ways to Exterminate Web Bugs

Overview

In This Chapter

- Avoiding markup and spelling faux pas
- Keeping links hot and fresh
- Gathering beta testers to check, double-check, and triple-check your site
- Applying user feedback to your site

After you put the finishing touches on a set of pages (but before you post them on the Web for the world to see) is the time to put them through their paces. Testing is the best way to control your site's quality.

Thorough testing *must*include content review, analysis of HTML syntax and semantics, link checks, and various sanity checks to make doubly sure that what you built is what you really wanted. Read this chapter for some gems of testing wisdom (learned from a lifetime of Web adventures) - as we seek to rid your Web pages of bugs, errors, gaucheries, and lurking infelicities.

Make a List and Check It - Twice

Your design should include a road map (often called a *site map*) that tells you what's where in every individual HTML document in your site, and the relationships among these pages. If you're really smart, you've kept this map up-to-date as you moved from design to implementation. (In our experience, things always change when you go down this path.) If you're merely as smart as the rest of us, don't berate yourself - update that map *now.* Be sure to include all intra- and inter-document links.

A site map provides the foundation for a test plan. Yep, that's right - effective testing isn't random. Use the map to

- Investigate and check every page and every link systematically.
- Make sure everything works as you think it should - and that what you built has some relationship (however surprising) to your design.
- Define the list of things to check as you go through the testing process.
- Check everything (at least) twice. (Red suit and reindeer harness optional.)

Master Text Mechanics

By the time any collection of Web pages comes together, you're looking at thousands of words, if not more. Yet many Web pages get published without even a cursory spell check, which is why we suggest - no, demand - that you include a spell check as a step when testing and checking your materials. (Okay, we don't have a gun to your head, but you *know* it's for your own good.) Many HTML tools, such as FrontPage, HomeSite, and Dreamweaver, include built-in spell checkers and that's the first spell-check method you should use. These HTML tools also know how to ignore the HTML markup and just check your text.

Even if you only use HTML tools occasionally and hack out the majority of your HTML by hand, perform a spell check before posting your documents to the Web. (For a handy illustration of why this step matters, try keeping a log of spelling and grammar errors you find during your Web travels. Be sure to include a note on how those gaffes reflect on the people who created the pages involved. Get the message?)

You can use your favorite word processor to spell check your pages. Before you check them, add HTML markup to your custom dictionary, and pretty soon, the spell checker runs more smoothly - getting stuck only on URLs and other strange strings that occur from time to time in HTML files.

If you'd prefer a different approach, try any of the many HTML-based spell-checking services now available on the Web. We like the one at the Doctor HTML site, which you can find online at `www2.imagiware.com/RxHTML/`.

If Doctor HTML's spell checker doesn't float your boat, visit a search engine, such as `www.yahoo.com`, and use `web page spell check` as a search string. Doing so can help you produce a list of spell-check tools made specifically for Web pages. Poke around `www.webreference.com` for more pointed pointers, if you feel so inclined!

One way or another, persist until you root out all typos and misspellings. Your users may not know to thank you for your impeccable use of language - but if they don't trip over any errors while exploring your work, they'll have a higher opinion of your pages (and of their creator). Even if they don't know why. Call it stealth diplomacy.

Lack of Live Links — A Loathsome Legacy

We performed an unscientific, random sample to double-check our own suspicions; users told us that positive impressions of a particular site are proportional to how many working links they find there. The moral of this survey: Always check your links. This step is as true after you publish your pages as it is before they're made public. Nothing irritates users more than a link that produces the dreaded `404 Server not found` error instead of the good stuff they seek! Remember, too, that link checks are as indispensable to page maintenance as they are to testing.

If you're long on twenty-first-century street smarts, hire a robot to do the job for you: They work really long hours (no coffee breaks), don't charge much, and faithfully check every last link in your site (or beyond, if you let them). The best thing about robots is that you can schedule them to do their jobs at regular intervals: They always show up on time, always do a good job, and never complain (though we haven't yet found one that brings homemade cookies or remembers birthdays). All you have to do is search online for words such as *link check* or *robot.* You'll find lots to choose from!

We're rather fond of a robot named MOMspider, created by Roy Fielding of the W3C. Visit the MOMspider site at `http://ftp.ics.uci.edu/pub/websoft/MOMspider/`. This spider takes some work to use, but you can set it to check only local links, and it does a bang-up job of catching stale links before users do. (Note that some HTML software, such as HomeSite, comes with a built-in link checker so you can check your links before you post your pages.)

If a URL points to one page that immediately points to another (a pointer), you're not entitled to just leave the link alone. Sure, it technically works, but for how long? And how annoying! So if your link checking shows a pointer that points to a pointer (yikes), do yourself (and your users) a favor by updating the URL to point *directly* to the content's real location. You save users time, reduce bogus traffic on the Internet, and generate good cyberkarma.

If you must leave a URL active even after it has become passé to give your users time to bookmark your new location, you can instruct newer browsers to jump straight from the old page to the new one by including the following HTML command in the old document's `<head>` section:

```
<meta http-equiv="refresh" content="0"; url="newurlhere">
```

This nifty line of code tells a browser (if sufficiently new) that it should refresh the page. The delay before switching to the new page is specified by the value of the `content` attribute, and the destination URL by the value of the `url` attribute. If you must build such a page, be sure to include a plain-vanilla link in its `<body>` section, too, so users with older browsers can follow the link manually, instead of automatically. You may also want to add some text that tells the visitors to update their bookmarks with the new URL. Getting there may not be half the fun, but it's the whole objective.

Look for Trouble in All the Right Places

You and a limited group of users should test your site well before you share it with the rest of the world - and more than once. This process is called *beta testing,* and it's a bona fide, five-star *must* for a well-built Web site, especially if you intend it for business use. When the time comes to beta-test a site, bring in as rowdy and refractory a crowd as you can find. If you have picky customers (or colleagues who are pushy, opinionated, or argumentative), be comforted to know that you have found a higher calling for them: Such people make ideal beta-testers - if you can get them to cooperate.

Don't wait till the very last minute to test your Web site. Sometimes the glitches found during the beta-test phase can take weeks to fix. Take heed: Test early and test often, and you'll thank us in the long run!

Beta-testers use your pages in ways you never imagined possible. They interpret your content to mean things you never intended in a million years. They drive you crazy and crawl all over your cherished beliefs and principles. And they do all this before your users do! Trust us, it's a blessing in disguise.

These colleagues also find gotchas, big and small, that you never knew existed. They catch typos that word processors couldn't. They tell you things you left out and things that you should have omitted. They give you a fresh perspective on your Web pages, and they help you to see them from extreme points of view.

The results of all this suffering, believe it or not, are positive. Your pages emerge clearer, more direct, and more correct than they would have if you tried to test them by yourself. (If you don't believe us, of course, you *could* try skipping this step. And when real users start banging on your site, forgive us if we don't watch.)

Cover All the Bases with Peer Reviews

If you're a user with a simple home page or a collection of facts and figures about your private obsession, this particular tip may not apply to you. But feel free to read along anyway — it just might come in handy down the road.

If your pages express views and content that represent an organization, chances are oh, *about 100 percent* that you should subject your pages to some kind of peer-and-management review before publishing them to the world. In fact, we recommend that you build reviews into each step along the way as you build your site — starting by getting knowledgeable feedback on such basic aspects as the overall design, writing copy for each page, and the final assembly of your pages into a functioning site. These reviews help you avoid potential stumbling blocks, such as unintentional off-color humor or unintended political statements. If you have any doubts about copyright matters, references, logo usage, or other important details, get the legal department involved (if you don't have one, you may want to consider a little consulting help for this purpose).

Building a sign-off process into reviews so you can prove that responsible parties reviewed and approved your materials may be a good idea. We hope you don't have to be that formal about publishing your Web pages, but it's far, far better to be safe than sorry. (So is this process best called*covering the bases,* or covering, ah, something else? You decide.)

Use the Best Tools of the Testing Trade

When you grind through your completed Web pages, checking your links and your HTML, remember that automated help is available. If you visit the W3C HTML validator at `http://validator.w3.org`, you'll be well on your way to finding computerized assistance to make your HTML pure as air, clean as the driven snow, and standards-compliant as, ah, *really well-written HTML.* (Do we know how to mix a metaphor, or what?)

Likewise, investigating the Web spiders discussed earlier in the chapter is a good idea; use them regularly to check links on your pages. These faithful creatures tell you if something isn't current, so you know where to start looking for links that need fixing.

Foster Feedback

Even after you publish your site, testing never ends. (Are you having flashbacks to high school or college yet? We know we are.) You may not think of user feedback as a form (or consequence) of testing, but it represents the best reality-check that your Web pages are ever likely to get, which is why doing everything you can — including offering prizes or other tangible inducements — to get users to fill out HTML forms on your Web site is a good idea.

This reality-check is also why reading *all* the feedback you get is even better. Go out and solicit as much as you can handle (don't worry; you'll soon have more). But the best idea of all is to carefully consider the feedback that you read and then implement the ideas that actually bid fair to improve your Web offerings. Oh, and it's a really good idea to respond to feedback with personal e-mail, to make sure your users know you're reading what they're saying. If you don't have time to do so, make some!

Even the most finicky and picky of users can be an incredible asset: Who better to pick over your newest pages and to point out those small, subtle errors or flaws that they revel in finding? Your pages will have contributed mightily to the advance of human society by actually finding a legitimate use for the universal human delight in nitpicking. And your users develop a real stake in boosting the success of your site. Working with your users can mean that some become more involved in your work, helping guide the content of your Web pages (if not the rest of your professional or obsessional life). Who could ask for more? Put it this way: You may yet find out, and it could be remarkably helpful.

If You Give to Them, They'll Give to You!

Sometimes, simply asking for feedback or providing surveys for users to fill out won't produce the results you might be seeking - either in quality or in volume. Remember the old days when you'd occasionally get a dollar bill in the mail, to encourage you to fill out a survey? It's hard to deliver cold, hard cash via the Internet, but a little creative action on your part should make it easy for you to offer your users something of value in exchange for their time and input. It could be an extra month on a subscription, discounts on products or services, or some kind of freebie by mail. (Maybe now you can finally unload those stuffed Gila Monsters you bought for that trade show last year. . . .)

But there's another way you can give back to your users that might not even cost you too much. An offer to send participants the results of your survey, or to otherwise share what you learn, may be all the incentive participants need to take the time to tell you what they think, or to answer your questions. Just remember that you're asking your users to give of their time and energy, so it's only polite to offer something in return.

Schedule Site Reviews

Okay, you probably understand that every time you change or update your Web site, you should test its functionality, run a spell check, perform a beta test, and otherwise jump through important hoops to put your best foot forward online. But sometimes, you'll make just a small change — a new phone number or address, a single product listing, a change of name or title to reflect a promotion and so on — and you won't go through the whole formal testing process for "just one little thing."

That's perfectly understandable, but one thing inevitably leads to another, and so on. Plus, if you solicit feedback, the chances are good that you'll get something back that points out a problem you'd never noticed or considered before. It's essential to schedule periodic Web site reviews, even if you've made no big changes or updates since your last review. Information grows stale, things change, and tiny errors have a way of creeping in as one small change succeeds another.

Just as you take your car in for an oil change, or swap out your air-conditioning filter, you should plan to check out your Web site regularly. Most big organizations we talk to do this every three months or so; others do it more often. Even when you think you have no bugs to catch, errors to fix, or outdated information to refresh, you'll often be surprised by what a review turns up. Make this part of your routine, and your surprises will be less painful — and require less work to remedy!

Chapter 20: Ten HTML Do's and Don'ts

Overview

In This Chapter

- Concentrating on content
- Going easy on the graphics, bells, whistles, and hungry T. Rexes
- Creating well-formulated HTML and then testing, testing, testing
- Finishing the work after the building is done

By itself, HTML is neither excessively complex nor overwhelmingly difficult. As some high-tech wags (including a few rocket scientists) have put it, *HTML ain't rocket science!* Nevertheless, a few important do's and don'ts can make or break the Web pages you build with HTML. Consider these humble admonishments as guidelines to help you make the most of HTML without losing touch with your users or watching your page blow up on its launch pad.

If some fundamental points that we made throughout this book seem to crop up here too (especially regarding proper and improper use of HTML), it's no accident. Heed ye well the prescriptions and avoid ye the maledictions. But hey, we know they're your pages and you can do what you want with them. Your users will decide the ultimate outcome. (We'd *never* say, 'We told you so!' Nope. Not us.)

Never Lose Sight of Your Content

So we return to the crucial question of payload: the content of your page. Why? Well, as Darrell Royal (legendary football coach of the University of Texas Longhorns in the ‘60s and ‘70s) is rumored to have said to his players, “Dance with who brung ya.” In normal English (as opposed to Texan), we think this means that you should stick with the people who’ve supported you all along and give your loyalty to those who’ve given it to you.

We’re not sure what this means for football, but for Web pages it means keeping faith with your users and keeping content paramount. If you don’t have strong, solid, informative content, users quickly get that empty feeling that starts to gnaw when a Web page is content-free. Then they’ll be off to richer hunting grounds on the Web, looking for content.

To satisfy their hunger, place your most important content on your site’s major pages. Save the frills and supplementary materials for secondary pages. The short statement of this principle for HTML is, “Tags are important, but what’s between the tags — the content — is what really counts.” For a refresher course on making your content the best it can possibly be, take a spin through Chapter 2.

Structure Your Documents and Your Site

Providing users with a clear road map and guiding them through your content is as important for a single home page as it is for an online encyclopedia. When longer or more complex documents grow into a full-fledged Web site, a road map becomes even more important. This map ideally takes the form of (yep, you guessed it) a flow chart that shows page organization and links. If you like pictures with a purpose, the chart could appear in graphic form in an explicitly labeled 'site map.'

We're strong advocates of top-down page design: Don't start writing content or placing tags until you understand what you want to say and how you want to organize your material. Then start building your HTML document or collection of documents with paper and pencil (or whatever modeling tool you like best). Sketch out relationships within the content and among your pages. Know where you're building before you roll out the heavy equipment.

Good content flows from good organization. It helps you stay on track during page design, testing, delivery, and maintenance. And organization helps your users find their way through your site. Need we say more? Well, yes: Don't forget that *organization changes with time.* Revisit and critique your organization and structure on a regular basis, and don't be afraid to change either one to keep up with changes in your content or focus.

Keep Track of Those Tags

Although you're building documents, it's often easy to forget to use closing tags, even when they're required (for example, the `</a>` that closes the opening anchor tag `<a>`). Even when you're testing your pages, some browsers can be a little too forgiving; they compensate for your errors, leaving you with a false sense of security.

The Web is no place to depend on the kindness of strangers; scrutinize your tags to head off possible problems from other browsers that may not be quite so understanding (or lax, as the case may be).

As for the claims that some vendors of HTML authoring tools make ("You don't even have to know any HTML!"), all we can say is, *Uh-huh, surrre.* HTML itself is still a big part of what makes Web pages work; if you understand it, you can troubleshoot with minimum fuss. Also, ensuring that your page's inner workings are correct and complete is something only *you* can do for your documents, whether you build them yourself or a program builds them for you.

We could go on ad infinitum about this, but we'll exercise some mercy and confine our remarks to the most pertinent:

- **Keep track of tags yourself while you write or edit HTML by hand.** If you open a tag — be it an anchor, a text area, or whatever — create the closing tag for it right then and there, even if you have content to add. Most HTML editors do this for you.
- **Use a syntax checker to validate your work during the testing process.** Syntax checkers are automatic tools that find missing tags — and other ways to drive you crazy! Use these whether you build pages by hand or with software assistance. Here's the URL for the W3C's HTML validator: `http://validator.w3.org/`.
- **Obtain and use as many browsers as you can when testing pages.** This not only alerts you to missing tags; it can also point out potential design flaws or browser dependencies (covered in a section later in this chapter). This exercise also emphasizes the importance of alternate text information. That's why we always check our pages with Lynx (a character-only browser).
- **Always follow HTML document syntax and layout rules.** Just because most browsers don't require elements such as `<html>`,`<head>`, and `<body>` doesn't mean you can omit them; it just means that browsers don't give a hoot if you do. But browsers per se are not your audience. Your users (and, for that matter, future browsers) may indeed care.

Although HTML isn't exactly a programming language, it still makes sense to treat it like one. Following formats and syntax helps you avoid trouble, and careful testing and rechecking of your work ensures a high degree of quality, compliance with standards, and a relatively trouble-free Web site.

Make the Most from the Least

More is not always better, especially when it comes to Web pages. Try to design and build your pages using minimal ornaments and simple layouts.

Don't overload pages with graphics; add as many levels of headings as you can fit, and make sure your content is easy to read and follow. Make sure any hyperlinks you include add real value to your site.

Gratuitous links to useless information are nobody's friend; if you're tempted to link to a Web cam that shows a dripping faucet, resist, resist, resist!

Structure and images exist to *highlight* content. The more bells, whistles, and dinosaur growls dominate a page, the more distracted from your content visitors will be. Use structure and graphics sparingly, wisely, and as carefully as possible. Anything more can be an obstacle to content delivery. Go easy on the animations, links, and layout tags, or risk having your message (even your page) devoured by a hungry T. Rex.

Build Attractive Pages

When users visit Web pages with a consistent framework that focuses on content, they're likely to feel welcome. The important thing is to *supplement* content with graphics and links — not to trample users with an onslaught of pictures and links. Making Web pages pretty and easy to navigate only adds to a site's basic appeal, and makes your cyber-campers even happier.

If you need inspiration, cruise the Web and look for layouts and graphics that work for you. If you take the time to analyze what you like, you can work from other people's design principles without having to steal details from their layouts or looks (which isn't a good idea anyway).

As you design your Web documents, start with a basic, standard page layout. Pick a small but interesting set of graphical symbols or icons and adopt a consistent navigation style. Use graphics sparingly (yes, you've heard this before); make them as small as possible — limit size, number of colors, shading, and so on, while retaining eye appeal. When you build simple, consistent navigation tools for your site, label them clearly and use them everywhere. You can make your pages appealing, informative, and inviting if you invest the time and effort.

Avoid Browser Dependencies

When you're building Web pages, the temptation to view the Web in terms of your favorite browser is hard to avoid. That's why you should always remember that users view the Web in general (and your pages in particular) from many perspectives - through many different browsers.

During the design and writing phases, you'll probably ping-pong between HTML and a browser's-eye view of your work. At this point in the process, we recommend switching from one browser to another, testing your pages among a group of browsers (including at least one character-mode browser). This helps balance how you imagine your pages and helps keep you focused on content.

There are public telnet servers with Lynx (a character-mode browser) installed that you can use for free and don't require software installation. Visit `www.trill-home.com/lynx/public_lynx.html` for a good list of telnet servers featuring Lynx.

During testing and maintenance, you must browse your pages from many different points of view. Work from multiple platforms; try both graphical and character-mode browsers on each page. Such testing takes time but repays that investment with pages that are easy for everyone to read and follow. It also helps viewers who come at your materials from platforms other than your own, and helps your pages achieve true independence from any single viewpoint. Why limit your options?

If several pages on your site use the same basic HTML, create a single template for those pages. Test the template with as many browsers as you can find. When you're sure the template is browser-independent, use it to create other pages. This helps you ensure that every page looks good, regardless of which browser a visitor might be using, and puts you on your way to real HTML enlightenment.

Think Evolution, Not Revolution

Over time, Web pages change and grow. Keep a fresh eye on your work, and keep recruiting fresh eyes from the ranks of those who haven't seen your work before, to avoid what we call "organic acceptance."

This concept is best explained by the analogy of your face and the mirror: You see it every day; you know it intimately, so you aren't as sensitive as someone else to the impact of changes over time. Then you see yourself on video, or in a photograph, or through the eyes of an old friend. At that point, changes obvious to the world reveal themselves to you as you exclaim, "I've gone completely gray!" or "My spare tire could mount on a semi!" or "Who the heck is *that?*"

As with the rest of life, changes to Web pages are usually evolutionary, not revolutionary. They proceed with small daily steps; big radical leaps are rare. Nevertheless, you must stay sensitive to the supporting infrastructure and readability of your content as pages evolve. Maybe the lack of on-screen links to each section of your Product Catalog didn't matter when you had only three products — but now that you offer 25, it's a different story. You've heard that form follows function; in Web terms, the structure of your site needs to follow changes in its content. If you regularly reevaluate your site's effectiveness at communicating its contents, you'll know when it's time to make changes, whether large or small.

This is why user feedback is crucial. If you don't get feedback through forms or other means, go out and aggressively solicit info from your users. If you're not sure how you're doing, then consider: If you don't ask for feedback, how can you tell?

Navigating Your Wild and Woolly Web

Users who view the splendor of your site don't want to be told *you can't get there from here.* Aids to navigation are vital amenities on a quality Web site. In Chapter 9, for example, we introduce the concept of a*navigation bar* - a consistent graphical place to put buttons that help users get from A to B. By judicious use of links and careful observation of what constitutes a complete screen (or screenful) of text, you can help your users minimize (or even avoid) scrolling. Text anchors make it easy to move to the previous and or next screens, as well as to the top, index, and bottom in any document. Just that easy, just that simple, or so it appears to the user.

We believe pretty strongly in the *low scroll* rule: That is, users should have to scroll *no more than one* screenful in either direction from a point of focus or entry without finding a navigation aid to let them jump (not scroll) to the next point of interest.

We don't believe that navigation bars are required, or that the names for controls should always be the same. We do believe that the more control you give users over their reading, the better they like it. The longer a document gets, the more important such controls become; they work best if they occur about every 30 lines in longer documents (or in a separate, always-visible frame if you use HTML frames).

Beating the Two-Dimensional Text Trap

Conditioned by centuries of printed material and the linear nature of books, our mindsets can use an adjustment. The nonlinear potentials of hypermedia give the Web a new definition for the term *document*. Of course, it's tempting to pack your pages full of hyper-capabilities until it resembles a Pony Express dynamite shipment and gallops off in many directions at once. To avoid this, judge your hypermedia according to whether it (1) adds interest, (2) expands on your content, or (3) makes a serious — and relevant — impact on the user.

Within these constraints, such material can vastly improve any user's experience of your site.

Stepping intelligently outside old-fashioned linear thinking can improve your users' experience of your site and make your information more accessible to your audience. That's why we encourage careful use of document indexes, cross-references, links to related documents, and other tools to help users navigate within your site. Keep thinking about the impact of links as you look at other people's Web materials; it's the quickest way to shake free of the linear-text trap. (The printing press was high-tech for its day, but that *was* 500 years ago!) If you're looking for a model for your site's behavior, don't think about your new trifold four-color brochure, however eye-popping; think about how your customer-service people interact with new customers on the telephone. (*What can I do to help you today?*)

Overcome Inertia through Constant Vigilance

When you're dealing with your Web materials post-publication, remember that it's only human to goof off after finishing a big job. Maintenance may not be nearly as heroic, inspiring, or remarkable as creation, yet it represents most of the activity that's needed to keep any document alive and well. Sites that aren't maintained often become ghost sites; users stop visiting sites when developers stop working on them. Never fear - a little work and attention to detail keeps your pages fresh. If you start with something valuable and keep adding value, a site's value appreciates over time - just like any other artistic masterpiece. Start with something valuable and leave it alone, and it soon becomes stale and loses value.

Consider your site from the viewpoint of a master aircraft mechanic: Correct maintenance is a real, vital, and ongoing accomplishment, without which you risk a crash. A Web site, as a vehicle for important information, deserves regular attention; maintaining a Web site requires discipline and respect. See www.disobey.com/ghostsites/index.shtml for a humorous look at ghost sites.

Keeping up with constant change translates into creating (and adhering to) a regular maintenance schedule. Make it somebody's job to spend time on a site regularly; check to make sure the job's getting done. If someone is set to handle regular site updates, changes, and improvements, normally they start flogging other participants to give them things to do when scheduled site maintenance rolls around. Next thing you know, everybody's involved in keeping information fresh - just as they should be. This keeps your visitors coming back for more!

Part VII: Appendixes

In This Part:

Appendix A: HTML 4 Tags

Appendix B: HTML Character Codes

Appendix C: Glossary

In this part . . .

This part of the book supplements the main text with useful resources and summary information. It includes a complete list of HTML elements, with syntax and brief explanations, a reasonably complete listing of common HTML character codes, a glossary of technical terms found elsewhere in the book, and an index you can use to find your way around the text. We hope you'll not only get to know these supporting members of our cast, but that you'll also use them often and well!

Appendix A: HTML 4 Tags

In This Appendix

- Tracking HTML elements
- Getting a heads up on deprecated and empty tags
- Tracking tag attributes

Keeping track of the bountiful bevy of HTML elements can often be tricky, even for experienced Webmasters. To make your HTML life a bit easier, we created this appendix, which includes a table that lists the following information for each element:

- **Name:** The name of the element.
- **Chapter number:** The chapter(s) in which you can find information about the element.

  Some of the more complicated, less frequently used, or deprecated elements are not discussed in detail in this book. Please visit the W3C site at `www.w3.org/TR/html4/index/elements.html` for more information on these elements.
- **Empty:** If the tag is listed with the letter *E,* the tag is an empty tag and is forbidden to have an end tag.
- **Deprecated:** If the tag is listed with a *D,* then the tag is deprecated, and you should consider using a style sheet rule in its place.
- **Description:** A description of the tag.
- **Attributes:** This list includes the attributes that can be used with the element. When the element takes the common attributes, also called core attributes, we simply say "All core attributes." (As a reminder, those attributes are `id`, `class`, `style`, and `title`.)

  There are also language and event attributes that can be used with most elements. Likewise, when these attributes apply to elements, we simply say "language attributes" and/or "event attributes," respectively. The language attributes are `dir` and `lang`. The event attributes are `onclick`, `ondblclick`, `onmousedown`, `onmouseup`, `onmouseover`, `onmousemove`, `onmouseout`, `onkeypress`, `onkeydown`, and `onkeyup`. See Chapter 12 for more information on what these attributes do. If an element can only use some of these event attributes, we list them individually. If we just say "event attributes," the element uses all of them.

Table A-1: HTML Tags

Name	Ch #	Empty?	Deprecated	Description	Attributes
a	5, 13	N		Anchor	All core attributes, language attributes, event attributes, `accesskey, charset, coords, href, hreflang, name, onblur, onfocus, rel, rev, shape, tabindex, type`
abbr	7	N		Abbreviated form (for WWW, HTTP)	All core attributes, language attributes, event attributes

acronym	7	N		Indicates an acronym	All core attributes, language attributes, event attributes
address	N/A	N		Information on author	All core attributes, language attributes, event attributes
applet	N/A	N	D	Java applet	All core attributes, `align, alt, archive, class, code, codebase, height, hspace, id, name, object, style, title, vspace, width`
area	6	Y		Client-side image map area	All core attributes, language attributes, event attributes, `accesskey, alt, coords, href, nohref, onblur, onfocus, shape, tabindex`
b	7	N		Bold text style	All core attributes, language attributes, event attributes
base	N/A	Y		Document base URI	`href, target`
basefont	N/A	Y	D	Base font size	`color, face, id, size`
bdo	N/A	N		I18N BiDi override	All core attributes, language attributes
big	7	N		Large text style	All core attributes, language attributes, event attributes
blockquote	4	N		Long quotation	All core attributes, language attributes, event attributes, `cite`
body	4, 7	N		Document body	All core attributes, language attributes, event attributes, `alink, background, link, onload, onunload, text, vlink`
br	4	Y		Forced line break	All core attributes

button	N/A	N		Push button	All core attributes, language attributes, event attributes, `accesskey, disabled, name, onblur, onfocus, tabindex, type, value`
caption	N/A	N		Table caption	All core attributes, language attributes, event attributes
center	8	N	D	Use`div align="center"` instead	
cite	7	N		Citation	All core attributes, language attributes, event attributes
code	7	N		Computer code fragment	All core attributes, language attributes, event attributes
col	8	Y		Table column	All core attributes, language attributes, event attributes, `align, char, charoff, span, valign, width`
colgroup	8	N		Table column group	All core attributes, language attributes, event attributes, `align, char, charoff, span, valign, width`
dd	4	N		Definition description	All core attributes, language attributes, event attributes
del	7	N		Deleted text	All core attributes, language attributes, event attributes, `cite, datetime`
dfn	7	N		Instance definition	All core attributes, language attributes, event attributes
dir	N/A	N	D	Directory list	All core attributes, language attributes, event attributes, `compact`
div	11	N		Generic language/ style container	All core attributes, language attributes, event attributes

dl	4	N		Definition list	All core attributes, language attributes, event attributes
dt	4	N		Definition term	All core attributes, language attributes, event attributes
em	7	N		Emphasis	All core attributes, language attributes, event attributes
fieldset	N/A	N		Form control group	All core attributes, language attributes, event attributes, `accesskey, align`
font	7	N	D	Local change to font	All core attributes, language attributes, `face, color, size`
form	10	N		Interactive form	All core attributes, language attributes, event attributes, `accept, acceptcharset, action, enctype, method, name, onreset, onsubmit`
frame	9	Y		Subwindow	All core attributes, `frameborder, longdesc, marginheight, marginwidth, name, noresize, scrolling, src`
frameset	9	N		Window subdivision	All core attributes, `cols, onload, onunload, rows`
h1-h6	4	N		Heading 1-6	All core attributes, language attributes, event attributes
head	4	N		Document head	language attributes, `profile`
hr	4	Y		Horizontal rule	All core attributes, language attributes, event attributes
html	4	N		Document root element	Language attributes
i	7	N		Italic text style	All core attributes, language attributes, event attributes

iframe	N/A	N		Inline subwindow	All core attributes, `align`, `frameborder`, `height`, `longdesc`, `marginheight`, `marginwidth`, `name`, `scrolling`, `src`, `width`
img	6	Y		Embedded image	All core attributes, language attributes, event attributes, `alt`, `height`, `ismap`, `longdesc`, `name`, `src`, `usemap`, `width`
input	10, 12, 15	Y		Form control	All core attributes, language attributes, event attributes, `accept`, `accesskey`, `alt`, `checked`, `disabled`, `ismap`, `maxlength`, `name`, `onblur`, `onchange`, `onfocus`, `onselect`, `readonly`, `size`, `src`, `tabindex`, `type`, `usemap`, `value`
ins	7	N		Inserted text	All core attributes, language attributes, event attributes, `cite`, `datetime`
isindex	N/A	Y	D	Single line prompt	All core attributes, language attributes, `prompt`
kbd	7	N		Text to be entered by the user	All core attributes, language attributes, event attributes
label	N/A	N		Form field label text	All core attributes, language attributes, event attributes, `accesskey`, `onblur`, `onfocus`

legend	N/A	N		Fieldset legend	All core attributes, language attributes, event attributes, `accesskey`
li	4	N		List item	All core attributes, language attributes, event attributes
link	11	Y		A media-independent link	All core attributes, language attributes, event attributes, `charset, href, hreflang, media, rel, rev, type`
map	6	N		Client-side image map	All core attributes, language attributes, event attributes, `name`
menu	N/A	N D		Menu list	All core attributes, language attributes, event attributes, `compact`
meta	4	Y		Generic meta-information	Language attributes, `http-equiv, name, content, scheme`
noframes	9	N		Alternate content container for non rendering	All core attributes, language attributes, event attributes
noscript	N/A	N		Alternate content container for non script-based rendering	All core attributes, language attributes, event attributes
object	13	N		Generic embedded object	All core attributes, language attributes, event attributes, `archive, classid, codebase, codetype, data, declare, height, name, standby, tabindex, type, usemap, width`
ol	4	N		Ordered list	All core attributes, language attributes, event attributes

optgroup	N/A	N		Option group	All core attributes, language attributes, event attributes, `disabled, multiple, name, onblur, onchange, onfocus, size, tabindex`
option	10	N		Selectable choice	All core attributes, language attributes, event attributes, `disabled, multiple, name, onblur, onchange, onfocus, size, tabindex`
p	4	N		Paragraph	All core attributes, language attributes, event attributes, `align`
param	13	Y		Named property value	`id, name, type, value, valuetype`
pre	4	N		Preformatted text	All core attributes, language attributes, event attributes, `width`
q	N/A	N		Short inline quotation	All core attributes, language attributes, event attributes, `cite`
s	7	N	D	Strikethrough text style	All core attributes, language attributes, event attributes
samp	7	N		Sample program output, scripts, and so on	All core attributes, language attributes, event attributes
script	12	N		Script statements	`charset, defer, language, src, type`
select	10	N		Option selector	All core attributes, language attributes, event attributes, `disabled, multiple, name, onblur, onchange, onfocus, size, tabindex`
small	7	N		Small text style	All core attributes, language attributes, event attributes

span	11	N		Generic language/style container	All core attributes, language attributes, event attributes
strike	7	N	D	Strikethrough text	All core attributes, language attributes, event attributes
strong	7	N		Strong emphasis	All core attributes, language attributes, event attributes
style	11	N		Style info	language attributes, `media, title, type`
sub	N/A	N		Subscript	All core attributes, language attributes, event attributes
sup	N/A	N		Superscript	All core attributes, language attributes, event attributes
table	8	N		Table	All core attributes, language attributes, event attributes, `align, border, cellpadding, cellspacing, frame, rules, summary, width`
tbody	8	N		Table body	All core attributes, language attributes, event attributes, `align, char, charoff, valign`
td	8	N		Table data cell	All core attributes, language attributes, event attributes, `abbr, align, axis, char,charoff, colspan, headers, rowspan, scope, valign`
textarea	10	N		Multi-line text field	All core attributes, language attributes, event attributes, `accesskey, cols, disabled, name, onblur, onchange, onfocus, onselect, readonly, rows, tabindex`

tfoot	8	N		Table footer	All core attributes, language attributes, event attributes, `align, char, charoff, valign`
th	8	N		Table header cell	All core attributes, language attributes, event attributes, `abbr, align, axis, char, charoff, colspan, headers, rowspan, scope, valign`
thead	8	N		Table header	All core attributes, language attributes, event attributes, `align, char, charoff, valign`
title	4	N		Document title	Language attributes
tr	8	N		Table row	All core attributes, language attributes, event attributes, `align, bgcolor, char, charoff, valign`
tt	7	N		Teletype or monospaced text type	All core attributes, language attributes, event attributes
u	7	N D		Underlined text style	All core attributes, language attributes, event attributes
ul	4	N		Unordered list	All core attributes, language attributes, event attributes
var	7	N		Instance of a variable or program argument	All core attributes, language attributes, event attributes

Appendix B: HTML Character Codes

Overview

In This Appendix

- Knowing your Latin and Greek characters
- Formatting for special punctuation
- Handling mathematical characters
- Dealing with ISO character sets

This appendix includes listings for all the character sets supported by HTML 4.0 (and consequently HTML 4.01). Use the nine tables here to find out what you have to type in order to get HTML to output characters that aren't part of the ASCII character set. Each table includes:

- The character
- The character's numeric entity
- The character's character entity

Chapter 1 details why you might want to use any of these entities and how you use them with your markup.

Keep in mind that we couldn't get some characters to display correctly because of font restrictions, so be sure to try using the entity and displaying it in your browser to see what symbol appears.

Even though everything listed here is part of the HTML 4.01 standard, *not all browsers support these characters.* Even if you test your code on your favorite browser before you consider it ready for action, such testing is no guarantee of universal usability. As a general rule, the character shortcuts are least supported and the numeric characters are most supported.

ISO-Latin-1

ISO-Latin-1 is fully supported by all current and most older browsers. Review Table B-1 to find out what you need to know. Check out Table B-9 in the section "More ISO Character Sets Than You Can Shake a Stick At," later in this appendix.

Table B-1: The ISO-Latin-1 Character Set

Character/Description	Numeric Entity	Character Entity
Em space, not collapsed	None	
En space	None	
Em space	&#00; - &#08;	
Horizontal tab		
Line feed or new line
Unused	&#11; - &#31;	
Space		
!	!	
"	"	"
#	#	
$	$	
%	%	
&	&	&
' (Apostrophe)	'	
(	(	
)	)	
*	*	
+	+	
, (Comma)	,	
– (Minus sign)	-	
. (Period)	.	
/	/	
0-9 (Numerals, where	0 - 9	None 0 is 0, 1 is 1, and so on)
:	:	
;	;	
<	<	<
=	=	
>	>	>
?	?	

@	@	
A-Z (Capitals, where	A - Z	None A is A, B is B, and so on)
[	[	
\	\	
]	]	
^	^	
_ (Underscore)	_	
` (Grave accent)	`	
a-z (Lowercase, where	a - z	None a is a, b is b, and so on)
{	{	
|			
}	}	
~	~	
Unused	&#127; - Ÿ	
Nonbreaking space		
¡	¡	¡
¢	¢	¢
£	£	£
€	¤	¤
¥	¥	¥
»	¦	¦
§	§	§
¨	¨	¨
©	©	©
ª	ª	ª
«	&#171	«
¬	¬	¬
- (Soft hyphen)		
®	®	®
¯	¯	¯
°	°	°
±	±	±
²	²	²
³	³	³

´ (Acute accent)	´	´
µ	µ	µ
¶	¶	¶
•	·	·
¸ (Cedilla)	¸	¸
1	¹	¹
º	º	º
»	»	»
1/4	¼	¼
1/2	½	½
3/4	¾	¾
¿	¿	¿
À	À	À
Á	Á	Á
Â	Â	Â
Ã	Ã	Ã
Ä	Ä	Ä
Å	Å	Å
Æ	Æ	Æ
Ç	Ç	Ç
È	È	È
É	É	É
Ê	Ê	Ê
Ë	Ë	Ë
Ì	Ì	Ì
Í	Í	Í
Î	Î	Î
Ï	Ï	Ï
a	Ð	Ð
Ñ	Ñ	Ñ
Ò	Ò	Ò
Ó	Ó	Ó
Ô	Ô	Ô
Õ	Õ	Õ
Ö	Ö	Ö
x	×	×

Ø	Ø	Ø
Ù	Ù	Ù
Ú	Ú	Ú
Û	Û	Û
Ü	Ü	Ü
´Y	Ý	Ý
I	Þ	Þ
ß	ß	ß
à	à	à
á	á	á
â	â	â
ã	ã	ã
ä	ä	ä
å	å	å
æ	æ	æ
ç	ç	ç
è	è	è
é	é	é
ê	ê	ê
ë	ë	ë
ì	ì	ì
í	í	í
î	î	î
ï	ï	ï
b	ð	ð
ñ	ñ	ñ
ò	ò	ò
ó	ó	ó
ô	ô	ô
õ	õ	õ
ö	ö	ö
÷	÷	÷
ø	ø	ø
ù	ù	ù
ú	ú	ú
û	û	û

ü	ü	ü
´y	ý	ý
I	þ	þ
ÿ	ÿ	ÿ

Greek Characters

If you have a need to use Greek characters, Table B-2 gives you everything you need to know.

Table B-2: Greek Characters

Character	Numeric Entity	Character Entity
Α	Α	Α
Β	Β	Β
Γ	Γ	Γ
Δ	Δ	Δ
Ε	Ε	Ε
Ζ	Ζ	Ζ
Η	Η	Η
Θ	Θ	Θ
Ι	Ι	Ι
Κ	Κ	Κ
Λ	Λ	Λ
Μ	Μ	Μ
Ν	Ν	Ν
Ξ	Ξ	Ξ
Ο	Ο	Ο
Π	Π	Π
Ρ	Ρ	Ρ
Σ	Σ	Σ
Τ	Τ	Τ
Υ	Υ	Υ
Φ	Φ	Φ
Χ	Χ	Χ
Ψ	Ψ	Ψ
Ω	Ω	Ω
α	α	α
β	β	β
γ	γ	γ
δ	δ	δ
ε	ε	ε
ζ	ζ	ζ
η	η	η

θ	θ	θ
ι	ι	ι
κ	κ	κ
λ	λ	λ
μ	μ	μ
ν	ν	ν
ξ	ξ	ξ
ο	ο	ο
π	π	π
ρ	ρ	ρ
ς	ς	ς
σ	σ	σ
τ	τ	τ
υ	υ	υ
φ	φ	φ
χ	χ	χ
ψ	ψ	ψ
ω	ω	ω
θ	ϑ	ϑ
ϒ	ϒ	ϒ
ϖ	ϖ	ϖ

Special Punctuation

Use the information in Table B-3 to help format bullets, horizontal ellipsis, and other special punctuation.

Table B-3: Special Punctuation

Character	Numeric Entity	Character Entity
•	•	•
...	…	…
'	′	′
"	″	″
‾	‾	‾
⁄	⁄	⁄

Characters That Could Be Letters

Maybe in a parallel universe, these characters might be letters. We're not sure why you'd need to use these characters, but they are laid out for you in Table B-4.

Table B-4: Letter-like Characters

Character	Numeric Entity	Character Entity
	℘	℘
ℑ	ℑ	ℑ
←	ℜ	ℜ
™	™	™
↵	ℵ	ℵ

Characters That Point

We never really knew how many ways you can make an arrow until we learned HTML. Now we pass on this wisdom to you in Table B-5.

Table B-5: Arrow Characters

Character	Numeric Entity	Character Entity
♦	←	←
	↑	↑
∅	→	→
¬	↓	↓
×	↔	↔
⌡	↵	↵
	⇐	⇐
	⇑	⇑
	⇒	⇒
	⇓	⇓
	⇔	⇔

Heavy-Duty Mathematical Characters

Get out your pocket protector. If you're in a technical or scientific field, the characters in Table B-6 might be right up your subset.

Table B-6: Mathematical Characters

Character	Numeric Entity	Character Entity
∀	∀	∀
	∂	∂
∃	∃	∃
	∅	∅
∆	∇	∇
	∈	∈
	∉	∉
∋	∋	∋
π	∏	∏
Σ	∑	∑
-	−	−
*	∗	∗
√	√	√
∝	∝	∝
8	∞	∞
	∠	∠
	ࢳ	∧
	ࢴ	∨
≈	∪	∪
	∫	∫
∴	∴	∴
~	∼	∼
≅	≅	≅
˜	≈	≈
	≠	≠
+	≡	≡
=	≤	≤
=	≥	≥
℘	⊂	⊂
	⊃	⊃

⊄	⊄	⊄
⊆	⊆	⊆
	⊇	⊇
	⊕	⊕
	⊗	⊗
⊥	⊥	⊥
	⋅	⋅

Technical Characters

The technical characters in Table B-7 are funky-looking brackets. Do with them what you will, and rest easy knowing that they exist.

Table B-7: Technical Characters

Character	Numeric Entity	Character Entity
⌈	⌈	⌈
⌉	⌉	⌉
⌊	⌊	⌊
	⌋	⌋
⟨	〈	⟨
⟩	〉	⟩

Hearts, Spades, Clubs, and Diamonds

Card fiend? Check out Table B-8.

Table B-8: Playing Card Symbols

Character	Numeric Entity	Character Entity
	♠	♠
⇓	♣	♣
♥	♥	♥
→	♦	♦

More ISO Character Sets Than You Can Shake a Stick At

Not that you would shake a stick at anything. Ever. Who shakes sticks? Anyway, in addition to the various characters shown in this appendix, numerous variants of the ISO-Latin character set have been created, primarily to support developers (and users) who want to read Web pages in languages other than English. AsTable B-9 shows, there are 10 named versions of the ISO-Latin character sets, and 15 versions of ISO-Latin itself, each of which is aimed at a separate collection of languages. If you want to service readers in languages other than English, these character sets will be important to you.

Table B-9: ISO 8859 Character Sets

Character Set	Script	Languages
ISO-8859-1	Latin-1	ASCII plus most Western European languages, including Albanian, Afrikaans, Basque, Catalan, Danish, Dutch, English, Faroese, Finnish, Flemish, Galician, German, Icelandic, Irish, Italian, Norwegian, Portuguese, Scottish, Spanish, and Swedish. Omits certain Dutch, French, and German characters.
ISO-8859-2	Latin-2	ASCII plus most Central European languages, including Czech, English, German, Hungarian, Polish, Romanian, Croatian, Slovak, Slovene, and Serbian.
ISO-8859-3	Latin-3	ASCII plus characters required for English, Esperanto, German, Maltese, and Galician.
ISO-8859-4	Latin-4	ASCII plus most Baltic languages, including Latvian, Lithuanian, German, Greenlandic, and Lappish; now superseded by ISO-Latin-6.
ISO-8859-5		ASCII plus Cyrillic characters for Slavic languages, including Byelorussian, Bulgarian, Macedonian, Russian, Serbian, and Ukrainian.
ISO-8859-6		ASCII plus Arabic characters.
ISO-8859-7		ASCII plus Greek characters.
ISO-8859-8		ASCII plus Hebrew.

ISO-8859-9	Latin-5	Latin-1 except that some Turkish symbols replace Icelandic ones.
ISO-8859-10	Latin-6	ASCII plus most Nordic languages, including Latvian, Lithuanian, Inuit, non-Skolt Sami, and Icelandic.
ISO-8859-11		ASCII plus Thai.
ISO-8859-12	Latin-7	ASCII plus Celtic.
ISO-8859-13	Latin-8	ASCII plus the Baltic Rim characters.
ISO-8859-14	Latin-9	ASCII plus Sami (Finnish).
ISO-8859-15	Latin-10	Variation on Latin-1 that includes Euro currency sign, plus extra accented Finnish and French characters.

Appendix C: Glossary

A

absolute

When used to modify pathnames or URLs, a full and complete file specification (as opposed to a relative one). An absolute specification includes a host identifier, a complete volume, and path specification.

anchor

In HTML, an anchor is tagged text or a graphic element that acts as a link to another location inside or outside a given document, or an anchor may be a location in a document that acts as the destination for an incoming link. The latter definition is most commonly how we use it in this book.

animation

A computerized process of creating moving images by rapidly advancing from one still image to the next. In HTML, animated GIFs are typically used to produce this effect.

attribute

A named characteristic associated with a specific HTML element. Some attributes are required, and others are optional. Some attributes also take values (if so, the syntax is `attribute="value"`), depending on the particular element and attribute involved.

B

bandwidth

Technically, the range of electrical frequencies a device can handle; more often, bandwidth is used as a measure of a communication technology's carrying capacity. The more bandwidth users have on their machines, the more quickly they can access HTML files.

beta testing

When you and a limited group of users test your Web site before you share it with the rest of the world.

bookmark

A reference from a saved list of URLs kept by the Netscape Web browser. Bookmarks allow quick loading of a Web site without retyping the URL. Bookmarks are also known as Favorites in Microsoft Internet Explorer.

browser

A Web access program that can request HTML documents from Web servers and render such documents on a user's display device. See also ***client.***

bugs

Issues that sometimes show up in software in the form of major or minor errors, mistakes, and gotchas. Bugs got their name from insects found in antiquated tube-based computers of the late 1950s and early 1960s that were attracted to the glow of the filament in a tube. Bugs that show up in HTML markup can mess up a visitor's attempt to navigate your site.

C

CGI (Common Gateway Interface)

The specification that governs how Web browsers communicate with and request services from Web servers; also the format and syntax for passing information from browsers to servers via HTML forms or document-based queries.

client

The end-user side of the client/server arrangement; typically, client refers to a consumer (rather than a provider) of network services; a Web browser is therefore a client program that talks to Web servers.

client-side image map

The same as a server-side image map, except that the hot-spot definitions are stored within the HTML document on the client side, rather than in a map file stored on the server.

content

The raison d'être for HTML; although form is important, content is why users access Web documents and why they come back for more.

CSS (Cascading Style Sheets)

A method of markup that allows users to define how certain HTML, XHTML, or XML structural elements, such as paragraphs and headings, should be displayed using style rules instead of additional markup. The versions of CSS are CSS1 and CSS2. CSS2 is the most recent completed version, and CSS3 is underway.

D

default

In general computer-speak, a selection made automatically in a program if the user specifies no explicit selection. For HTML, the default is the value assigned to an attribute if none is supplied.

deprecated

The term used to earmark an HTML element or attribute that is to be left for dead by future versions of HTML.

DOCTYPE declaration

HTML markup that tells the processor where to locate the DTD and contains declarations for the particular document. Also called a *document type declaration*.

document

The basic unit of HTML information; a document refers to the entire contents of any single HTML file. Because this definition doesn't always correspond to normal notions of a document, we refer to what can formally be called HTML documents more or less interchangeably with Web pages, which is how browsers render such documents for display.

DTD (Document Type Definition)

A formal SGML specification for a document. A DTD lays out the structural elements and markup definitions to be used to create instances of documents.

E

element

A section of a document defined by a start and end tag or an empty tag.

e-mail

An abbreviation for electronic mail; e-mail is the preferred method for exchanging information between users on the Internet (and other networked systems).

empty tag

An HTML element that does not require the use of a closing tag. In fact, the use of a closing tag in empty tags is forbidden.

entity

A character string that represents another string of characters.

error message

Information delivered by a program to a user, usually to inform him or her that the process hasn't worked properly, if at all. Error messages are an ill-appreciated art form and contain some of the funniest and most opaque language we've ever seen (also, the most tragic for their unfortunate recipients).

event

A user activity, such as moving a mouse pointer over an image, clicking a link, or selecting a drop-down menu.

external style sheet

A style sheet that resides outside a Web document in a separate, external file.

F

footer

The concluding part of an HTML document, containing contact, version, date, and attribution information to help identify a document and its authors. Most people use the `<address>` element to identify this information.

form handler

A program on the Web server or even possibly a simple `mailto` URL that manages the data a user sends to you via the form.

forms

HTML markup that lets browsers solicit data from users and then deliver that data to specially designated input-handling programs on a Web server. Briefly, forms provide a mechanism to let users interact with servers on the Web.

FTP (File Transfer Protocol)

An Internet file transfer service based on the TCP/IP protocols that provides a way to copy files to and from FTP servers elsewhere on a network.

G

GIF (Graphic Interchange Format)

A graphics format commonly used in Web documents because of its relatively small file size and relatively sharp resolution.

graphics

Files in HTML documents that belong to one of a restricted family of types (usually .GIF or .JPG). Graphics are referenced via URLs for inline display on Web pages.

GUI (Graphical User Interface)

Pronounced “gooey,” GUIs make graphical Web browsers possible; they create a visually oriented interface that makes it easy for users to interact with computerized information of all kinds.

H

heading

A markup element used to add document structure to HTML documents. Sometimes the term refers to the initial portion of an HTML document between the `<head> . . . </head>` tags, where titles and context definitions are commonly supplied.

helper applications

Applications that help a browser deliver Web information to users. Although recent browsers can display multiple graphics files (and sometimes other kinds of data), sometimes they must pass certain files — for instance, motion picture or sound files — over to other applications that know how to render the data they contain.

hexadecimal

A numbering system used to condense binary numbers. The hexadecimal system is composed of six letters and ten numbers. In HTML, hexadecimal numbering is used with elements and their attributes to denote what colors should comprise backgrounds and other elements in a Web page.

HTML (Hypertext Markup Language)

The SGML-derived markup language used to create Web pages. Not quite a programming language, HTML nevertheless provides a rich lexicon and syntax for designing and creating useful hypertext documents for the Web.

HTTP (Hypertext Transfer Protocol)

The Internet protocol used to manage communication between Web clients (browsers) and servers.

hyperlink

A shorthand term for *hypertext link,* a block of text that a user can activate with a mouse click to navigate to another Web page.

hypermedia

Any of a variety of computer media — including text, graphics, video, sound, and so on — available through hypertext links on the Web.

hypertext

A method of organizing text, graphics, and other kinds of data for computer use that lets individual data elements point to one another; a nonlinear method of organizing information, especially text.

I

image map

A synonym for *clickable image,* an overlaid collection of pixel coordinates for a graphic that a user can select to activate a related hypertext link for further Web navigation.

inline content

A word or string of words inside a block element.

inline element

Any element that controls presentation on an element-by-element basis, and an inline element does not denote structure. In other words, it's a text element, for example the <b> element is an inline element.

internal style sheet

A style sheet that resides inside the Web document in which you're working and controls how its information appears on-screen.

Internet

A worldwide collection of networks that began with technology and equipment funded by the U.S. Department of Defense in the 1970s. The World Wide Web is just a portion of the Internet. Today, it links users in nearly every country, speaking nearly every known language.

ISP (Internet Service Provider)

An organization that provides individuals or other organizations with access to the Internet, usually for a fee. ISPs usually offer a variety of communications options for their customers, ranging from analog telephone lines, to a variety of higher-bandwidth leased lines, to ISDN and other digital communications services.

J

Java

An object-oriented, platform-independent, secure, and compact programming language designed for Web application deployment. Most system vendors support Java, which was created by Sun Microsystems.

JPEG (or JPG)

Joint Photographic Experts' Group; an industry association that defined a highly compressible format for images designed for complex color still images (such as photographs). JPEG files take the extension .JPG or .JPEG. Today, .JPG is one graphics format of choice for Web use, particularly for complex or photographic images.

L

layout

The overall arrangement of the elements in a document.

link

Also called a *hyperlink.* A pointer in one part of an HTML document that can transport users to another part of the same document, or to another document entirely. This capability puts the*hyper* into hypertext. In other words, a link is a one-to-one relationship/association between two concepts or ideas.

Lynx

A widely used text-based Web browser, useful for checking a Web page before *going live* with it (or using a Web server to host the page on the Web).

M

maintenance

The process of regularly inspecting, testing, and updating the contents of Web pages; also, an attitude that such activities are both inevitable and advisable.

mark up

To embed special characters (*metacharacters*) within a text file to tell a computer program how to handle the file's contents.

markup language

A formal set of special characters and related capabilities used to define a specific method for handling the display of files that include markup; HTML is a markup language used to design and create Web pages.

metadata

Specially defined elements that describe a document's structure, content, or rendering within the document itself or through external references. (Metadata literally means data about data.)

multimedia

A method of combining text, sound, graphics, and full-motion or animated video within a single compound computer document.

N

nameservers

Computers on the Internet that translate domain names into the actual Internet location for your computer's browser.

navigation

Refers to the use of hyperlinks to move within or between HTML documents and other Web-accessible resources.

navigation bar

An element on a Web page that arranges a series of hypertext links on a single line to provide a set of navigation controls to help users move through an HTML document or a set of HTML documents.

nesting

One structure that occurs within another; in HTML, nesting happens most commonly with list structures that may be freely nested within one another, regardless of type.

O

online

A term that indicates information, activity, or communications located on or taking place in, an electronic, networked computing environment (such as the Internet).

operating system

The underlying control program on a computer that makes the hardware run and supports the execution of one or more applications. DOS, Windows, Unix, and OS/2 are all examples of operating systems.

P

page

The generic term for an HTML document that Web users view on their browsers.

PDF (Portable Document Format)

The rich, typographically correct document format of Adobe, used to provide multiplatform document access through its Acrobat software as a more powerful alternative to HTML.

plug-in

Hardware or software added to a system that adds a specific feature such as plug-ins that allow Netscape Navigator to play video.

properties

In CSS, they are the different aspects of the display of text and graphics, such as font size or background color.

R

relative address

An abbreviated document address that may be combined with the <base> element to create a complete address or is the complete address for a local file found in the same directory.

resource

Any HTML document, capability, or other item or service available via the Web. URLs point to resources.

robot

A special Web-traveling program that wanders widely, following and recording URLs and related titles for future reference in search engines. Also called a *spider.*

S

script

A set of programming instructions that activate when an event that you define occurs.

scripting language

A special kind of programming language that a computer reads and executes at the same time (which means that the computer figures out what to do with the language when it appears in a document or at the time that it's used. JavaScript is a common scripting language associated with Web use).

search engine

A Web-based application that searches the contents of a database of available Web pages and other resources to provide information that relates to specific topics or keywords, which a user supplies.

search tools

Any of a number of programs that can permit HTML documents to become searchable, using the`<isindex>` element to inform the browser of the need for a search window, and behind-the-scenes indexing and anchoring schemes to let users locate particular sections of or items within a document.

selector

In CSS, identifies the element to which a style rule applies.

SGML (Standard Generalized Markup Language)

An ISO standard document definition, specification, and creation mechanism that makes platform and display differences across multiple computers irrelevant to the delivery and rendering of documents.

shareware

Software, available by various means, that users can run for free for a trial period. After that trial period expires, users must register and purchase the software, or they must discontinue its use.

specification

A formal document that describes the capabilities, functions, and interfaces for a specific piece of software, a markup language, or a communications protocol.

spider

A Web-traversing program that tirelessly investigates Web pages and their links, while storing information about its travels for inclusion in the databases typically used by search engines. Also called *Web spider, Webcrawler,search bot,* and *robot.*

style sheet

A file that holds the layout settings for a certain category of a document. Style sheets, like templates, contain settings for headers and footers, tabs, margins, fonts, columns, and more.

syntax

The rules that govern how HTML markup looks and behaves within HTML documents. The real syntax definition for HTML comes from the SGML Document Type Definition (DTD). Markup written with the correct syntax is likelier to run well.

syntax checker

A program that checks a particular HTML document's markup against the rules that govern its use; a recommended part of the testing regimen for all HTML documents.

T

tag

The formal name for a piece of HTML markup that signals a command of sorts (instructions), usually enclosed in angle brackets (< >).

template

Literally, a model to imitate. In HTML terms, a template describes the skeleton of a Web page, including the HTML for its heading and footer, and a consistent layout, and a set of navigation elements.

test plan

The series of steps and elements to follow when conducting a formal test of software or other computerized systems; we strongly recommend that you write - and use - a test plan as a part of your Web-publication process.

thumbnail

A miniature rendering of a graphical image, used as a link to the full-size version.

title

The text supplied to a Web page's title bar when displayed, used as data in many Web search engines.

U

Unix

The operating system of choice for the Internet community at large and the Web community, too, Unix offers the broadest range of tools, utilities, and programming libraries for Web server use.

URL (Uniform Resource Locator)

The primary naming scheme used to identify Web resources. URLs define the protocols to use, the domain name of the Web server where a resource resides, the port address to use for communication, and a directory path to access named Web files or resources.

URL-encoded text

Text encoded specifically for the purpose of passing information requests and URL specifications to Web servers from browsers. URL encoding replaces spaces with plus signs (+) and substitutes special hexadecimal codes for a range of otherwise-irreproducible characters. This method is used to pass document queries from browsers to servers.

user interface

The overall design of your site, including the way it looks, fits together, and works to provide access to information. Abbreviated as *UI.* See also ***GUI.***

V

valid

Markup that follows all the syntax rules defined in a DTD, allowing the document to pass through a validator program with no errors.

validation

The process of comparing a document to a set of document rules, in this context a DTD.

W

Web

Also called the *World Wide Web, WWW,* or *W3*. The complete collection of all Web servers available on the Internet, which comes as close to containing the “sum of human knowledge” as anything we’ve yet seen.

Web page

Synonym for an HTML document. In this book, we refer to Web pages as *sets* of related, interlinked HTML documents, usually produced by a single author or organization.

Web server

A computer, usually on the Internet, that hosts HTTP protocols and related Web-service software. See also ***Web-hosting provider.***

Web site

An addressed location, usually on the Internet, that provides access to the set of Web pages that correspond to the URL for a given site. A Web site consists of a Web server and a named collection of Web documents, both accessible through a single Web address. See also ***URL.***

Web-hosting provider

A company that provides space on Web servers for individuals or companies to host Web sites. See also ***ISP.***

well-formed document

An HTML document that adheres to the rules that make it easy for a computer to interpret.

white space

The breathing room on a page — parts of a display or document unoccupied by text or other visual elements. A certain amount of white space is essential to make documents attractive and readable.

WYSIWYG (What You See Is What You Get)

A term used to describe text editors or other layout tools (such as HTML authoring tools) that attempt to show on-screen what final, finished documents will look like.

X

XHTML (Extensible Hypertext Markup Language)

The reformulation of HTML 4.0 as an application of XML 1.0.

XML (Extensible Markup Language)

A system for defining, validating, and sharing document formats. Its main difference from HTML is that you can create your own elements.

Index

A

<a> element, 352
    attributes,352
    audio/video links, 237-238
    basic links, 81-82
    target attribute, 177-178

abbr attribute
    <td> element, 359
    <th> element, 359

<abbr> element, 128,352

abbreviations,128

absolute,375

absolute links, 83-84

accent marks, 20,22-24

accept attribute
    <form> element, 354
    <input> element, 355

acceptcharset attribute, 354

Access (Microsoft), 250-251

accessibility
    alternative text for images, 100-102
    Bobby utility, 321
    checklist for making an accessible site, 321
    frames,161,163,166
    resources available, 344-345
    text-only Web browsers, 16
    Web Accessibility Guidelines, 321
    Web Accessibility Initiative (WAI), 102

accesskey attribute
    <a> element, 352
    <area> element, 352
    <button> element, 353
    <fieldset> element, 354
    <input> element, 355
    <label> element, 356
    <legend> element, 356
    <textarea> element, 359

<acronym> element, 128,352

acronyms,128

action attribute, 186-187,354

<address> element, 352

Adobe
    GoLive,217,281-282
    Photoshop,282-283

AIFF (Audio Interchange File Format), 233

align attribute

<applet> element, 352
<caption> element, 144
<col> element, 353
<colgroup> element, 353
<fieldset> element, 354
<hr> element, 69
<iframe> element, 355
<img> element, 105-106
<p> element, 61,357
<table> element, 150-151
<tbody> element, 359
<td> element, 151,359
<tfoot> element, 359
<th> element, 359
<thead> element, 359
<tr> element, 151,360

alignment
horizontal rule, 69
images,105-106
paragraphs,61
table captions, 144
table cells, 151-152
table rows, 151-152
tables,150-152

alink attribute, 119,353

alt attribute
<applet> element, 352
<area> element, 109,352
<img> element, 100,355
<input> element, 355

alternative text, 100-102

Amazon.com Web site, 322-323

anchor,375

anchor element, 81-82.*See also*<a> element

animation,375
animated GIFs, 111-112
Flash,235

Apple QuickTime Web site, 237

<applet> element, 352

archive attribute
<applet> element, 352
<object> element, 357

<area> element, 109,352

arrow character set, 370-371

ASP.NET,255-256

ASP.NET For Dummies, Bill Hatfield, 257

attributes (HTML markup), 20-22,375.*See also*individual attributes by name

audio
AIFF (Audio Interchange File Format), 233
AU (Sun/NeXT audio), 233
background music, 242

embedding files into Web pages, 239-241
links,237-239
MIDI (Musical Instrument Digital Interface), 233
MP3 (MPEG-1, Layer III), 233
QT (QuickTime), 233
RA (RealAudio), 233
RMF (Rich Music Format), 233-234
streaming audio, 242-244
SWA (Shockwave Audio), 234
SWF (Flash), 234
WAV (RIFF WAVE), 234
WMA (Windows Media Audio), 234

AVI (audio/video interleaved) video format, 234

`axis` attribute
`<td>` element, 359
`<th>` element, 359

Index

B

<b> element, 128,352

background (of Web page)
  Cascading Style Sheets (CSS), 205
  colors,120
  images,121–122
  music,242

background attribute, 121–122,353

bandwidth,375

<base> element, 352

<basefont> element, 352

BBEdit,275,280

<bdo> element, 352

beta testing, 330,375

bgcolor attribute
  <body> element, 120,122
  <td> element, 159–160
  <tr> element, 360

<bgsound> element, 242

<big> element, 128,352

<block> element, 59

block quotes, 63–64

<blockquote> element, 63–66,353

blocks of text, 60

Bobby utility for scanning Web site accessibility, 321,353

<body> element
  alink attribute, 119
  attributes,353
  background attribute, 121–122
  bgcolor attribute, 120,122
  content and markup, 58–59,118
  link attribute, 119
  text attribute, 118
  vlink attribute, 119

boldfacing text, 128

bookmark,375

border attribute
  <img> element, 104–105
  <table> element, 143,145

borders
  Cascading Style Sheets (CSS), 206
  frames,173–175
  images,104–105
  tables,138,143,145

 element, 67–68,353

Bravenet Web site, 347

broken links
- common causes of, 84–85
- finding, 304, 329
- link checkers, 284–285, 329

browsers, 375
- alternative text, 100–102
- Cascading Style Sheets (CSS), 159, 207–208
- character set support, 361
- client-side scripts, 223
- color settings, 119
- colors, 116
- display instructions, 12–13
- frames, 161, 166
- graphical browsers, 16
- Hypertext Transfer Protocol (HTTP), 17
- image display, 100
- interpretation of HTML, 15–16
- line breaks, 67
- Lynx, 16, 379
- opening a new window, 86–87
- page title display, 54
- plug-ins, 236
- redirecting users, 56–57
- scripting languages, 224
- streaming media, 242–243
- style sheets, 159, 203, 207–208
- testing, 16, 337, 339
- text-only browsers, 16
- viewing Web pages, 42–43
- white space, 64–65
- XHTML, 260, 263
- XML (Extensible Markup Language), 260

bugs, 376. *See also* errors

Builder.com Web site, 96, 257

bullet styles, 75–76

bulleted lists, 74–76

`<button>` element, 353

Index

C

`<caption>` element, 144,353

captions in tables, 144

card character set, 373

Cascading Style Sheets (CSS), 376
- background properties, 205
- border properties, 206
- browser display, 207–208
- browser support, 159
- classification properties, 206
- core style sheets, 215
- CSS1,204–205
- CSS2,204–205
- CSS3,205
- declarations,207,209
- external style sheet, 213–215,377
- font properties, 206
- inheritance,211–212
- internal style sheet, 213–214,379
- list properties, 206
- margin properties, 206
- padding properties, 206
- positioning properties, 206
- predefined style sheets, 215
- properties,207
- properties, defined, 381
- selector,381
- selectors,207,209
- size properties, 206
- style classes, 209–211
- style rules, 207,209
- table properties, 206
- tables,159
- text properties, 206
- validator,209
- values,207
- W3C,205

Cascading Style Sheets For Dummies, Damon A. Dean, 206

case-sensitivity
- HTML tags, 3
- URLs,84

`cellpadding` attribute, 148–149

cells (tables), 138
- alignment,151–152
- colors,159–160
- creating,140–142
- height,146,148
- padding,148–149
- populating,155–156
- spacing,148–150

span, 138, 142–143, 152–155
width, 138, 146–147

cellspacing attribute, 148–150

 element, 150, 353

CGI (Common Gateway Interface), 376

CGI Resource Index Web site, 197

CGI scripts for handling forms, 196–197

char attribute
<col> element, 353
<colgroup> element, 353

char attribute *(continued)*
<tbody> element, 359
<td> element, 359
<tfoot> element, 359
<th> element, 359
<thead> element, 359
<tr> element, 360

character sets
arrow character set, 370–371
browser support, 361
card character set, 373
Greek character set, 367–369
ISO 8859 character sets, 373–374
ISO-Latin-1 character set, 362–367
letter-like character set, 370
mathematical character set, 371–372
punctuation character set, 369
specifying with the <meta> element, 57
technical character set, 372–373

charoff attribute
<col> element, 353
<colgroup> element, 353
<tbody> element, 359
<td> element, 359
<tfoot> element, 359
<th> element, 359
<thead> element, 359
<tr> element, 360

charset attribute
<a> element, 352
<link> element, 356
<script> element, 358

check boxes (forms), 189–190

checked attribute, 355

citations, 128

cite attribute
<blockquote> element, 353
<del> element, 354
<ins> element, 355
<q> element, 358

<cite> element, 128, 353

`class` attribute, 210–211,352

`classid` attribute, 357

“click here” phrasing in links, 319

clickable images. *See*image maps

client,376

client-side image maps, 376

client-side scripts, 223

clip art, 113

code, denoting in text, 129

`code` attribute, 352

`<code>` element, 129,353

code examples
  companion Web site, 7
  typing in, 2

`codebase` attribute
  `<applet>` element, 352
  `<object>` element, 357

`codetype` attribute, 357

`<col>` element, 144,353

`<colgroup>` element, 144,353

`color` attribute
  `<basefont>` element, 352
  `<font>` element, 125,127

colors
  background color, 120
  browser display, 116
  converters (RGB notation to hexadecimal notation), 117
  coordination tips, 119,122–123
  defining,116,120–121
  hexadecimal codes, 116–118
  hexadecimal color codes, 378
  links,119–120
  names,116
  table cells, 159–160
  text color, 118,125
  Web browser settings, 119

`cols` attribute
  `<frameset>` element, 167–170,355
  `<textarea>` element, 359

`colspan` attribute
  `<td>` element, 142–143,152–154,359
  `<th>` element, 359

columns.*See also*cells(tables)
  frames,167–172
  tables,144

Common Gateway Interface (CGI), 376

`compact` attribute
  `<dir>` element, 354
  `<menu>` element, 356

compressing images, 96–97

content,376
- checking and updating, 304
- frames,172–173
- images,94
- importance of, 335–336
- inline content, 379
- inline element, 379
- types of, 11–12

`content` attribute, 55,356

converting colors (RGB notation to hexadecimal notation), 117

`coords` attribute
- <a> element, 352
- `<area>` element, 109,352

copyright of images, 114

core style sheets, 215

Corel WordPerfect 8.0 for Windows, 281

crawlers,307

CSS (Cascading Style Sheets), 376
- background properties, 205
- border properties, 206
- browser display, 207–208
- browser support, 159
- classification properties, 206
- core style sheets, 215
- CSS1,204–205
- CSS2,204–205
- CSS3,205
- declarations,207,209
- external style sheet, 213–215,377
- font properties, 206
- inheritance,211–212
- internal style sheet, 213–214,379
- list properties, 206
- margin properties, 206
- padding properties, 206
- positioning properties, 206
- predefined style sheets, 215
- properties,207,381
- selectors,207,209,381
- size properties, 206
- style classes, 209–211
- style rules, 207,209
- table properties, 206
- tables,159
- text properties, 206
- validator,209
- values,207
- W3C,205

CuteFTP,300

Index

D

`data` attribute, 357

data collection forms, 181,183–185

databases,245
- ASP.NET,255–256
- design guidelines, 251
- drivers,256
- features,249
- instructions,252,254
- JSP,255–256
- linking to a Web page, 246–248
- Microsoft Access, 250–251
- Microsoft SQL Server, 251–252
- MySQL,249–250
- online catalogs, 247
- Oracle,251–252
- PHP,253–254
- queries,247,252–254
- SQL (Structured Query Language), 246–247
- Sybase,251–252
- technical support, 256–257
- user interaction, 29,246
- uses,245–246
- Web server connections, 256

`datetime` attribute
- `<del>` element, 354
- `<ins>` element, 355

`<dd>` element, 76–77

declarations (CSS), 207,209

`declare` attribute, 357

dedicated Internet connections, 293–294

default,376

`defer` attribute, 358

definition lists, 76–77

`<del>` element, 129,354

deprecated,376

description of page, 54–56

DevGuru Web site, 206

`<dfn>` element, 129,354

`<dir>` element, 354

Director (Macromedia), 234

`disabled` attribute
- `<button>` element, 353
- `<input>` element, 355
- `<optgroup>` element, 357
- `<option>` element, 357

<select> element, 358
<textarea> element, 359

 element, 354

<dl> element, 76–77,354

Doctor HTML spell checker, 328

DOCTYPE declaration, 376
HTML,52
XML,267

document,376
footer,377
header,53
structure,51–52
title,53–54

Document Type Definition (DTD), 377
HTML 4.01 specification, 25–26
XML,262–263

domain names, 297–300

download links, 90–91

download time
images,102–103
1-second, 1K rule, 98

downloading plug-ins, 236

Dreamweaver (Macromedia), 217,274–277

drop-down lists (forms), 187,192–193

drop-down menus, 33

<dt> element, 76–77,354

DTD (Document Type Definition), 377
HTML 4.01 specification, 25–26
XML,262–263

Index

E

ECMAscript, 224

editing HTML
- existing Web pages, 43–45
- HTML Tidy, 268
- word processors, 281

editors
- HTML editors, 273–278, 280–282
- image editors, 282–283
- text editors, 12, 36, 272–273

elements, 377. *See also specific elements*
- `<a>`, 81–82, 177–178, 237–238, 352
- `<abbr>`, 128, 352
- `<acronym>`, 128, 352
- `<address>`, 352
- anchor, 81–82
- `<applet>`, 352
- `<area>`, 109, 352
- `<b>`, 128, 352
- `<base>`, 352
- `<basefont>`, 352
- `<bdo>`, 352
- `<bgsound>`, 242
- `<big>`, 128, 352
- `<block>`, 59
- `<blockquote>`, 63–66, 353
- `<body>`, 58–59, 118–122, 353
- `<br>`, 67–68, 353
- `<button>`, 353
- `<caption>`, 144, 353
- case-sensitivity, 3
- `<center>`, 150, 353
- `<cite>`, 128, 353
- `<code>`, 129, 353
- `<col>`, 144, 353
- `<colgroup>`, 144, 353
- `<dd>`, 76–77
- `<del>`, 129, 354
- `<dfn>`, 129, 354
- `<dir>`, 354
- `<div>`, 354
- `<dl>`, 76–77, 354
- `<dt>`, 76–77, 354
- `<em>`, 129, 354
- `<embed>`, 239–242, 244
- empty, 21, 377
- end tags, 20, 336–337
- `<fieldset>`, 354
- `<font>`, 123–127
- `<form>`, 186–187, 354
- `<frame>`, 173–177, 354

<frameset>, 166–170, 355
<head>, 53–54
<h1>, 62, 355
<hr>, 68–70, 355
<h6>, 62, 355
<html>, 53, 355
<i>128, 355
<iframe>, 355
<img>98–109, 111, 355
inline, 379
<input>, 187–190, 194–195, 355
<ins>, 129, 355
<isindex>, 356
<kbd>, 129, 356
<label>, 356
<legend>, 356
, 72–75, 356
<link>, 213–215, 356
<map>, 109, 356
<menu>, 356
<meta>, 55–57
<noframes>, 166, 356
<noscript>, 357
<object>, 239–240, 357
, 72–74, 357
<optgroup>, 357
<option>, 187, 192–193, 357
<p>, 60–61, 357
<param>, 357
phrase, 128–131
<pre>, 65–66, 357
<q>, 358
<s>, 128, 358
<samp>, 129, 358
<script>, 221–222, 358
<select>, 187, 192–193, 358
singleton tag, 98
<small>, 128, 358
<span>, 358
start tag, 20
<strike>, 128, 358
<strong>, 129, 358
<style>, 213–214, 358
style, 128–131
style rules, 215
<sub>, 358
<sup>, 358
<table>, 140–143, 145–146, 148–151, 358–359
tag pair, 20
<tbody>, 144, 359
<td>, 140–143, 146, 148, 151–154, 159–160, 359
<textarea>, 187, 193, 359
<tfoot>, 144, 359
<th>, 144, 359
<thead>, 144, 359
<title>, 54, 359

<tr>,140–141,151–152,360
<tt>,128,360
<u>,128,360
,74–76,360
<var>,129,360

,129,354

e-mail,377
form data collection, 197
links to e-mail addresses, 91
Post Office Protocol (POP), 17
Simple Mail Transfer Protocol (SMTP), 17
spam,91

<embed> element, 239–242,244

embedding
audio/video files into a Web page, 239–241
database instructions into a Web page, 252–254
Windows Media Player, 241–242

empty element (or tag), 21,377

enctype attribute, 354

end tags, 20,336–337

entities (HTML markup), 20,22–24,377

error message, 377

errors,377
beta testing, 330
broken links, 84–85,284,329
404 Object Not Found error, 284
page redirectors, 330
peer reviews, 331
spelling errors, 328–329
validation of HTML, 331

events,377
attributes,220–221
scripts,219–220

expanding Web sites, 305–306,339–340

Extensible Markup Language (XML), 384
browser support, 260
defined,384
differences from HTML, 260
differences from XHTML, 260
DOCTYPE declaration, 267
Document Type Definition (DTD), 262–263
FAQ Web site, 262
how it works, 261–262
namespaces,267–268
rules,265–268
uses,261

external style sheets (CSS), 214–215,377

Index

F

`face` attribute
- `<basefont>` element, 352
- `<font>` element, 123–124,127

feedback from users, 331–332

Fetch,289

`<fieldset>` element, 354

file download links, 90–91

file formats for graphics, 95–97

file size of images, 96–98

File Transfer Protocol (FTP), 17,289,377

file transfers
- via FTP, 300–301
- via hosting provider's Web site, 301

file upload fields (forms), 191

FileMaker Pro 6 Bible, Steven A. Schwartz, 258

Fireworks (Macromedia), 283

FirstGov Web site, 17–18

Flash
- animation,235
- plug-in,235
- SWF (Shockwave File Extension), 234

Flash MX Bible, Robert Reinhardt and Snow Dowd, 235

`<font>` element
- `color` attribute, 125,127
- `face` attribute, 123–124,127
- `size` attribute, 125–127

font size
- absolute font sizes, 126–128
- Cascading Style Sheets (CSS), 206
- headings,62
- relative font sizes, 126–128
- setting,125–128

footer
- document footer, 377
- table footer, 144

`<form>` element, 354
- `action` attribute, 186–187
- attributes,354
- `method` attribute, 186

form handlers, 377

form validation, 229–230

formatting text
- phrase elements, 128–131
- style elements, 128–131

forms,377
check boxes, 189–190
creating,185–186
data collection forms, 181,183–185
degree of difficulty, 33
design tips, 198–199
drop-down lists, 187,192–193
e-mail collection of data, 197
file upload fields, 191
form handlers, 186,195–197,377
hidden fields, 190–191
input controls, 187–195
password fields, 188–189
radio buttons, 189–190
reset buttons, 194–195
search forms, 181–184
structure of, 186
submit buttons, 194–195
text areas, 187,193–194
text fields, 187–188
types of, 181
usability,198
uses,181–182
validating,229–230

`404 Object Not Found` error, 284

`<frame>` element, 354
attributes,354
`frameborder` attribute, 173–174
`marginheight` attribute, 174–175
`marginwidth` attribute, 174–175
`name` attribute, 177
`scrolling` attribute, 175–176

`frameborder` attribute
`<frame>` element, 173–174,354
`<iframe>` element, 355

frames
accessibility issues, 161,163,166
borders,173–175
browser support, 161,166
columns,167–172
combining rows and columns, 170–172
content,172–173
creating,164–165
frameset document, 165–166
height,167–168
HTML Frameset, 25,52
layout,164–165
links,177–179
margins,174–175
naming,177
navigation,164
nesting,179–180
rows,167–172
rules,173
scroll bars, 175–176

splitting, 179–180
target frame, 177–178
usability issues, 163
uses, 161–162, 164
width, 167–168

frameset document, 165–166

`<frameset>` element
attributes, 355
`cols` attribute, 167–170
description, 355
hierarchy among other elements, 166
`rows` attribute, 167–170

FrontPage (Microsoft), 277–278

FTP (File Transfer Protocol), 17, 289, 377

FTP clients, 289–290, 300–301

Index

G

ghost sites, 341

GIF Construction Set animation software, 111

GIF (Graphics Interchange Format), 95,97,378

GifBuilder animation software, 111

GoLive (Adobe), 217,281–282

graphic file formats, 95–97

Graphical User Interface (GUI), 378

graphical Web browsers, 16

graphics,378.*See also*images

graphics applications
- Adobe Photoshop, 282–283
- Jasc PaintShop Pro, 283
- Macromedia Fireworks, 283

Graphics Interchange Format (GIF), 95,97,378

Greek character set, 367–369

GUI (Graphical User Interface), defined, 378

Index

H

`<head>` element, 53-54

headers
- page,53
- table,144

`headers` attribute
- `<td>` element, 359
- `<th>` element, 359

headings,61-63,378

`height` attribute
- `<applet>` element, 352
- `<iframe>` element, 355
- `<img>` element, 102-103,355
- `<object>` element, 357
- `<table>` element, 148
- `<td>` element, 148

help resources, 345

helper applications, 378

hexadecimal color codes, 116-118,378

hidden fields (forms), 190-191

HomeSite,275,278-279

`<h1>` element, 62,355

horizontal rule, 68-71

hosting Web sites
- domain names, 297-300
- free Web hosting, 295
- with a hosting provider, 291,294-297
- on your own, 291-294

`<hr>` element, 68-70,355

`href` attribute
- `<a>` element, 83,91,352
- `<area>` element, 109,352
- `<base>` element, 352
- `<link>` element, 215,356

`hreflang` attribute
- `<a>` element, 352
- `<link>` element, 356

`<h6>` element, 62,355

`hspace` attribute
- `<applet>` element, 352
- `<img>` element, 106-107

`.htm` suffix, 42

HTML,378

HTML editors
- Adobe GoLive, 281-282
- BBEdit,280

Corel WordPerfect 8.0 for Windows, 281
Dreamweaver (Macromedia), 274-277
FrontPage (Microsoft), 277-278
functionality to look for, 273-274
helper editors, 273
HomeSite,278-279
Microsoft Word 97/2000 for Windows, 281
word processors, 281
WYSIWYG editors, 273

`<html>` element, 53,355

HTML elements. *See*elements

HTML forms, 377
check boxes, 189-190
creating,185-186
data collection forms, 181,183-185
degree of difficulty, 33
design tips, 198-199
drop-down lists, 187,192-193
e-mail collection of data, 197
file upload fields, 191
form handlers, 186,195-197,377
hidden fields, 190-191
input controls, 187-195
password fields, 188-189
radio buttons, 189-190
reset buttons, 194-195
search forms, 181-184
structure of, 186
submit buttons, 194-195
text areas, 187,193-194
text fields, 187-188
types of, 181
usability,198
uses,181-182
validating,229-230

HTML 4.01 specification, 25-26

HTML Frameset, 25,52

HTML Goodies Web site, 348

HTML help Web site, 345

HTML markup, 380
attributes,20-22
browser interpretation, 15-16
`DOCTYPE` declaration, 52
document structure, 51-53
elements,20-21
entities,20,22-24,377
examples,13-14
help resources, 345
limitations,204,259
testing on browsers, 16
tips for entering, 157
validating,286-289,331,337

HTML page, 380

HTML Strict, 25,52
`.html` suffix, 42
HTML Tidy, 268
HTML Transitional, 25,52
HTML Writers Guild, 346
HTTP (Hypertext Transfer Protocol), 17,378
`http://`,85
`httpequiv` attribute, 56-57,356
hyperlinks,378-379
    absolute links, 83-84
    active link, 119
    anchor element, 81-82
    broken links, 84-85,304,329
    'click here' phrasing, 319
    colors,119-120
    creating a basic hyperlink, 81-82
    creating a hyperlink that opens in a new window, 86-87
    creating a hyperlink to a specific location on another Web page, 89
    creating a hyperlink within a document, 87-88
    descriptive text, 319
    e-mail addresses, 91
    file downloads, 90-91
    frames,177-179
    how they work, 17
    image maps, 110
    images,99-100,107-108
    intradocument hyperlinks, 87-89
    link checkers, 284-285,329
    `mailto` links, 91
    multimedia,237-239
    off-site links, 318-319
    relative links, 83-84
    states,119
    underline,120
    URL,81-84
    visited link, 119
hypermedia,378
hypertext,378
Hypertext Markup Language (HTML), 378
Hypertext Transfer Protocol (HTTP), 17,378

Index

I

<i> element, 128,355

id attribute
- <applet> element, 352
- <basefont> element, 352
- <param> element, 357

<iframe> element, 355

image editing
- Adobe Photoshop, 282-283
- Jasc PaintShop Pro, 283
- Macromedia Fireworks, 283

image maps, 379
- <area> element, 109
- client-side image maps, 376
- creating,108-111
- <img> element, 109
- links,110
- <map> element, 109

image rollovers, 224-227

images
- alignment,105-106
- alternative text, 100-102
- animated GIFs, 111-112
- background image, 121-122
- border,104-105
- clip art, 113
- compression,96-97
- content,94
- copyright,114
- digital formats, 113-114
- download time, 102-103
- file size, 96-98
- GIFs,95,97
- graphic file formats, 95-97
- inserting,98-100
- JPEGs,96-97
- links,99-100,107-108
- look and feel, 95
- navigation,94,107-108
- optimizing,97-98
- PNGs,96-97
- size,102-104
- sources of low-cost and free images, 113-114
- spacing,106-107
- stock photography, 113
- thumbnails,110,382
- TIFFs,113-114
- tiling,121
- transparent images, 112
- uses,93-95

`<img>` element, 355
- `align` attribute, 105-106
- `alt` attribute, 100-101
- animated GIFs, 111
- attributes,355
- `border` attribute, 104-105
- `height` attribute, 102-103
- `hspace` attribute, 106-107
- image maps, 108-109
- singleton tag status, 98
- `src` attribute, 99-100,111
- `usemap` attribute, 109
- `vspace` attribute, 106-107
- `width` attribute, 102-103

Impact Online Web site, 28

inline content, 59,379

input controls (forms)
- check boxes, 189-190
- creating,187
- drop-down lists, 187,192-193
- file upload fields, 191
- hidden fields, 190-191
- password fields, 188-189
- radio buttons, 189-190
- reset buttons, 194-195
- submit buttons, 194-195
- text areas, 187,193-194
- text fields, 187-188

`<input>` element, 355
- attributes,355
- `type` attribute, 187-190,194-195

`<ins>` element, 129,355

inserting
- images,98-100
- paragraphs,60-61
- scripts into Web pages, 221

interactivity with databases, 29,246

interface.*See*user interface

Internal Revenue Service (IRS) Web site, 322,324

internal style sheets (CSS), 213-214,379

Internet,379

Internet connections (dedicated), 293-294

Internet protocols
- File Transfer Protocol (FTP), 17,289,377
- Hypertext Transfer Protocol (HTTP), 17,378
- Post Office Protocol (POP), 17
- Simple Mail Transfer Protocol (SMTP), 17

Internet Service Provider (ISP), 379

intradocument hyperlinks, 87-89

IRS Web site, 322,324

`<isindex>` element, 356

`ismap` attribute
    `<img>` element, 355
    `<input>` element, 355
ISO 8859 character sets, 373-374
ISO-Latin-1 character set, 362-367
ISP (Internet Service Provider), 379
italicizing text, 128
iTunes MP3 player, 236

Index

J

Jakob Nielsen's Useit Web site, 344

Jasc PaintShop Pro, 283

Java,379

JavaScript
- browser support, 225
- form validation, 229-230
- image rollovers, 224-227
- pop-up windows, 227-229
- sources of scripts, 225

JavaScript City Web site, 225

JavaScript For Dummies, Emily A. Vander Veer, 217

JavaScript Kit Web site, 225

JavaScript Source Web site, 225

Java-Scripts.Net Web site, 225

JavaServer Pages For Dummies, Mac Rinehart, 257

JPEG (Joint Photographic Experts Group), 96-97,379

JSP,255-256

Index

K

<kbd> element, 129,356

keywords
    metadata,55
    search engines, 308

Kira's Web Toolbox Web site, 347

Krug, Steve, usability expert, 344

Index

L

`<label>` element, 356

`language` attribute, 358

language options, 57

layout,379

`<legend>` element, 356

`<li>` element, 72–75,356

line breaks, 67–68

linear versus nonlinear approach to design, 340–341

lines (rules), 68–71

`link` attribute, 119,353

link checkers, 284–285,329

`<link>` element, 213–215,356

linking
- database to a Web page, 246–248
- domain name to a Web site, 299

links,378–379
- absolute links, 83–84
- active link, 119
- anchor element, 81–82
- broken links, 84–85,304,329
- “click here” phrasing, 319
- colors,119–120
- creating a basic link, 81–82
- creating a link that opens in a new window, 86–87
- creating a link to a specific location on another Web page, 89
- creating a link within a document, 87–88
- descriptive text, 319
- e-mail addresses, 91
- file downloads, 90–91
- frames,177–179
- how they work, 17
- image maps, 110
- images,99–100,107–108
- intradocument links, 87–89
- link checkers, 284–285,329
- `mailto` links, 91
- multimedia,237–239
- off-site links, 318–319
- relative links, 83–84
- states,119
- underline,120
- URL,81–84
- visited link, 119

lists
- bulleted lists, 74–76
- Cascading Style Sheets (CSS), 206
- definition lists, 76–77

drop-down lists (forms), 187,192–193
nested lists, 77–79
numbered lists, 72–74
ordered lists, 72–74
types of, 71
unordered lists, 74–76
uses,71

`longdesc` attribute
`<frame>` element, 354
`<iframe>` element, 355
`<img>` element, 355

`loop` attribute, 242

Lynx Web browser, 16,379

Index

M

Macromedia
- Director,234
- Dreamweaver,217,274-277
- Fireworks,283
- Flash,235
- Web site, 235,277

Macromedia Flash MX For Dummies, Gurdy Leete and Ellen Finkelstein, 235

`mailto` links, 91

maintaining Web sites, 301-305,341,380

`<map>` element, 109,356

mapping out a Web site, 311-315

`marginheight` attribute
- `<frame>` element, 174-175,354
- `<iframe>` element, 355

margins
- Cascading Style Sheets (CSS), 206
- frames,174-175

`marginwidth` attribute
- `<frame>` element, 174-175,354
- `<iframe>` element, 355

markup languages, 380

markup with HTML, 380
- attributes,20-22
- browser interpretation, 15-16
- `DOCTYPE` declaration, 52
- document structure, 51-53
- elements,20-21
- entities,20,22-24,377
- examples,13-14
- help resources, 345
- limitations,204,259
- testing on browsers, 16
- tips for entering, 157
- validating,286-289,331,337

mathematical character set, 371-372

Matt's Script archive Web site, 197

`maxlength` attribute, 355

`media` attribute
- `<link>` element, 356
- `<style>` element, 358

Mega Web Tools Web site, 347

`<menu>` element, 356

`<meta>` element, 55-57

metadata,54-55,380
- keywords,55-56

page description, 55-56
redirecting users, 56-57
search engines, 55-56

`method` attribute, 186,354

Microsoft
Access,250-251
FrontPage,277-278
SQL Server, 251-252
Word97/2000 for Windows, 281

MIDI (Musical Instrument Digital Interface), 233

MOMSpider link checker, 285,329

MPEG-1, Layer III (MP3), 233

MPG (Motion Picture Experts Group) video format, 234

MP3 (MPEG-1, Layer III), 233

MP3 players, 236

multimedia,380
audio formats, 233-234
background music, 242
bandwidth requirements, 231
degree of difficulty, 33
embedding files into Web pages, 239-241
links,237-239
players,235-237,241-242
pros and cons of using, 320
standards,231
streaming media, 242-244
uses,232
video formats, 234-235

`multiple` attribute
`<optgroup>` element, 357
`<option>` element, 357
`<select>` element, 358

music.*See*audio

Musical Instrument Digital Interface (MIDI), 233

MySQL,249-250

MySQL Bible, Steve Suehring, 258

MySQL Weekend Crash Course, Jay Greenspan, 258

MySQL: Your Visual Blueprint for Creating Open Source Databases, Michael Moncur, 258

MySQL/PHP Database Applications, 2nd Edition, Jay Greenspan and Brad Bulger, 258

Index

N

name attribute
- <a> element, 87,352
- <applet> element, 352
- <button> element, 353
- <form> element, 354
- <frame> element, 177,354
- <iframe> element, 355
- <img> element, 355
- <input> element, 355
- <map> element, 356
- <meta> element, 55,356
- <object> element, 357
- <optgroup> element, 357
- <option> element, 357
- <param> element, 357
- <select> element, 358
- <textarea> element, 359

nameservers,380

naming Web pages, 41–42

navigation,380
- frames,164
- home page, 317
- images,94,107–108
- intradocument links, 87–89
- navigation bar, 162,340,380
- rollovers,227
- scrolling,340
- site maps, 314–315
- testing,317–318
- topical navigation, 315–317
- user interface, 309

nesting,380
- elements (HTML markup), 21
- frames,179–180
- lists,77–79
- <pre> element inside <blockquote> element, 65–66
- tables,158

Nielsen, Jakob, usability expert, 324, 344

<noframes> element, 166,356

nohref attribute, 352

nonlinear versus linear approach to design, 340–341

noresize attribute, 354

<noscript> element, 357

noshade attribute, 70

Notepad,12

numbered lists, 72–74

Index

O

object attribute, 352

<object> element, 239–240,357

 element, 72–74,357

onabort attribute, 220

onblur attribute
- <a> element, 352
- <area> element, 352
- <button> element, 353
- how it works, 221
- <input> element, 355
- <label> element, 356
- <optgroup> element, 357
- <option> element, 357
- <select> element, 358
- <textarea> element, 359

onchange attribute
- how it works, 221
- <input> element, 355
- <optgroup> element, 357
- <option> element, 357
- <select> element, 358
- <textarea> element, 359

onclick attribute, 221

ondblclick attribute, 221

onerror attribute, 220

1-second, 1K rule, 98

onfocus attribute
- <a> element, 352
- <area> element, 352
- <button> element, 353
- how it works, 221
- <input> element, 355
- <label> element, 356
- <optgroup> element, 357
- <option> element, 357
- <select> element, 358
- <textarea> element, 359

onkeydown attribute, 221

onkeypress attribute, 221

onkeyup attribute, 221

online,380

online catalogs, 247

online training resources, 346

onload attribute
- <body> element, 353
- <frameset> element, 355

how it works, 220
onmousedown attribute, 221
onmousemove attribute, 220
onmouseout attribute, 221
onmouseover attribute, 221
onmouseup attribute, 221
onreset attribute
<form> element, 354
how it works, 221
onselect attribute
how it works, 221
<input> element, 355
<textarea> element, 359
onsubmit attribute
<form> element, 354
how it works, 221
onunload attribute
<body> element, 353
<frameset> element, 355
how it works, 220
opening a new window, 86–87
operating system, 380
<optgroup> element, 357
optimizing images, 97–98
<option> element, 187,192–193,357
Oracle,251–252
ordered lists, 72–74

Index

P

<p> element, 60-61,357

page,380

page description, 54-56

page header, 53

page redirectors, 56-57,330

page refresh, 330

page titles, 308,382

PaintShop Pro (Jasc), 283

pair of tags, 20

paragraphs,60-61

<param> element, 357

password fields (forms), 188-189

PDF (Portable Document Format), 380

PDImages.com Web site, 114

peer reviews, 331

Photoshop (Adobe), 282-283

Photoshop All-in-One Desk Reference For Dummies, Barbara Obermeier, 283

PHP,253-254

PHP and MySQL For Dummies, Janet Valade, 258

PHP Bible, 2nd Edition, Tim Converse and Joyce Park, 258

phrase elements, 128-131

players for multimedia, 235-237,241-242

plug-ins,381
- downloading,236
- Flash,235

PNG (Portable Network Graphics), 96-97

pointers,370-371

POP (Post Office Protocol), 17

populating tables, 155-156

pop-up text screens for alternative text, 100-102

pop-up windows
- opening a new window, 86-87
- scripting,227-229

Portable Document Format (PDF), 380

Portable Network Graphics (PNG), 96-97

Post Office Protocol (POP), 17

posting Web pages, 45-47

<pre> element, 65-66,357

predefined style sheets (CSS), 215

preformatted text, 64-66

programming
interactive Web sites, 30
scripts,218-219

Progressive Networks, 236

progressive video streaming, 243

promoting Web sites, 307-308

properties,381

protocols
File Transfer Protocol (FTP), 17,289,377
Hypertext Transfer Protocol (HTTP), 17,378
Post Office Protocol (POP), 17
Real Time Streaming Protocol (RTSP), 243
Simple Mail Transfer Protocol (SMTP), 17

publishing Web pages, 45-47

punctuation character set, 369

Index

Q

<q> element, 358

queries to a database, 247,252-254

QuickTime (QT)
audio,233
functionality,236-237
movies,234-235

quotes
<blockquote> element, 63-64
<q> element, 358

Index

R

radio buttons (forms), 189-190

RateGenius Web site, 199

`readonly` attribute
- `<input>` element, 355
- `<textarea>` element, 359

Real Time Streaming Protocol (RTSP), 243

Real Video (RV) video format, 235

RealAudio (RA), 233

RealNetworks,236

RealOne Player, 236

real-time video streaming, 243

redirecting users, 56-57,330

refreshing pages, 330

registering
- domain names, 299-300
- Web sites with search engines, 307

`rel` attribute
- `<a>` element, 352
- `<link>` element, 356

relative address, 381

relative links, 83-84

reset buttons (forms), 194-195

resources,381

`rev` attribute
- `<a>` element, 352
- `<link>` element, 356

Rich Music Format (RMF), 233-234

robots,381

rollovers
- degree of difficulty, 33
- navigation,227
- scripting,219-220,224-227

rows
- frames,167-172
- tables,140-142

`rows` attribute
- `<frameset>` element, 167-170,355
- `<textarea>` element, 359

`rowspan` attribute
- `<td>` element, 153-154,359
- `<th>` element, 359

RTSP (Real Time Streaming Protocol), 243

rules (lines), 68-71

RV (Real Video) video format, 235

Index

S

<s> element, 128,358

<samp> element, 129,358

saving
- Web pages, 41
- Word documents as text files, 38

scheme attribute, 356

scope attribute
- <td> element, 359
- <th> element, 359

<script> element, 221–222,358

scripting languages, 224,381

scripts,381
- Adobe GoLive, 217
- CGI scripts, 196–197
- client-side scripts, 223
- event attributes, 220–221
- events,219–220
- form validation, 229–230
- inserting into Web pages, 221
- Macromedia Dreamweaver, 217
- pop-up windows, 227–229
- programming functionality, 218–219
- rollovers,219–220,224–227
- server-side scripts, 223
- sources of scripts, 225
- uses,217–218

ScriptSearch Web site, 197

scroll bars (frames), 175–176

scrolling, use of in navigation, 340

scrolling attribute
- <frame> element, 175–176,354
- <iframe> element, 355

search engines, 381
- alternative text (for indexing images), 100
- keywords,308
- metadata,54–56
- page titles, 54
- registering your Web site, 307

search forms, 181–184

search tools, 381

<select> element, 187,192–193,358

selectors (CSS), 207,209,381

server-side scripts, 223

SGML (Standard Generalized Markup Language), 260,381

shading horizontal rules, 70

`shape` attribute
    `<a>` element, 352
    `<area>` element, 109,352

shareware,381

Shockwave Audio (SWA), 234

shopping carts, 33

Shoutcast multimedia streaming service, 236

Simple Mail Transfer Protocol (SMTP), 17

SimpleText,12

singleton tag, 98

site maps, 311–315,327–328,336

site reviews, 332

`size` attribute
    `<basefont>` element, 352
    `<font>` element, 125–127
    `<hr>` element, 69
    `<input>` element, 355
    `<optgroup>` element, 357
    `<option>` element, 357
    `<select>` element, 358

`<small>` element, 128,358

SMTP (Simple Mail Transfer Protocol), 17

Sonique MP3 player, 236

sound.*See*audio

spacing of images, 106–107

spam,91

`span` attribute
    `<col>` element, 353
    `<colgroup>` element, 144,353

`<span>` element, 358

specification,382

spell check, 328–329

spider,382

splitting frames, 179–180

SQL (Structured Query Language), 246–247

SQL For Dummies, Allen G. Taylor, 257

SQL Server (Microsoft), 251–252

SQL Weekend Crash Course, Allen G. Taylor, 258

`src` attribute
    `<bgsound>` element, 242
    `<frame>` element, 354
    `<iframe>` element, 355
    `<img>` element, 99,111,355
    `<input>` element, 355
    `<script>` element, 358

Standard Generalized Markup Language (SGML), 260,381

standards

HTML 4 for Dummies, 4th Edition

by Ed Tittel and Natanya Pitts

ISBN:0764519956

John Wiley & Sons © 2003 (408 pages)

Whether your goal is to build a simple, text-oriented Web site or one loaded with frames, graphics, and animation, this step-by-step book will put you on the right track.

Table of Contents

HTML 4 For Dummies, 4th Edition

Introduction

Part I - Meeting HTML in Its Natural Environment

Chapter 1 - The Least You Need to Know about HTML and the Web

Chapter 2 - HTML at Work on the Web

Chapter 3 - Creating Your First HTML Page

Part II - Getting Started with HTML

Chapter 4 - Structuring Your HTML Documents

Chapter 5 - Linking to Online Resources

Index

T

`tabindex` attribute
- `<a>` element, 352
- `<area>` element, 352
- `<button>` element, 353
- `<input>` element, 355
- `<object>` element, 357
- `<optgroup>` element, 357
- `<option>` element, 357
- `<select>` element, 358
- `<textarea>` element, 359

`<table>` element, 358
- `align` attribute, 150–151
- attributes,358–359
- `border` attribute, 143,145
- `cellpadding` attribute, 148–149
- `cellspacing` attribute, 148–150
- `height` attribute, 148
- hierarchy among other table elements, 140–142
- `width` attribute, 146

tables
- alignment,150–152
- body,144
- borders,138,143,145
- captions,144
- Cascading Style Sheets (CSS), 159
- cell padding, 148–149
- cell spacing, 148–150
- cell span, 138,142–143,152–155
- cell width, 138
- cells,138,140–142,146–147
- colors,159–160
- column groups, 144
- columns,144
- creating,140–142
- footer,144
- head,144
- headers,144
- height,146,148
- invisibility of, 135
- layout,138–140,144,159
- nesting,158
- populating,155–156
- rows,140–142
- standards,157
- table properties (CSS), 206
- testing,156
- uses,135–137
- white space, 148
- width,146–147

tag,382.*See also*elements

`target` attribute
  `<a>` element, 86,177–178
  `<base>` element, 352

`<tbody>` element, 144,359

`<td>` element, 359
  `align` attribute, 151
  attributes,359
  `bgcolor` attribute, 159–160
  `colspan` attribute, 142–143,152–154
  `height` attribute, 148
  hierarchy among other table elements, 140–141
  `rowspan` attribute, 153–154
  `valign` attribute, 151–152
  `width` attribute, 146

technical character set, 372–373

teletype text, 128

template,382

test plan, 382

testing
  beta testing, 330,375
  browsers,16,337,339
  navigation,317–318
  peer reviews, 331
  site maps, 328
  site reviews, 332
  style sheets, 203
  tables,156
  user feedback, 331–332

text
  abbreviations,128
  acronyms,128
  alternative text, 100–102
  big text, 128
  block quotes, 63–64
  blocks of text, 59–60
  boldface,128
  Cascading Style Sheets (CSS), 206
  citations,128
  colors,118,125
  computer code, display of, 129
  definition text, display of, 129
  deleted text, display of, 129
  emphasized text, display of, 129
  font face, 123–124
  font size, 125–128
  headings,61–63
  inline content, 59
  inserted text, display of, 129
  italics,128
  keyboard input text, display of, 129
  line breaks, 67–68
  link descriptions, 319
  paragraphs,60–61
  preformatted text, 64–66

sample output text, display of, 129
small text, 128
spell check, 328–329
strikethrough,128
teletype text, 128
underline text, 128
URL-encoded text, 383
variables, display of, 129
white space, 64–65
wrapping,67–68

text areas (forms), 187,193–194

`text` attribute, 118,353

text editors, 12,36,272–273

text fields (forms), 187–188

text size in headings, 62

text style elements, 128–131

`<textarea>` element, 187,193,359

text-only Web browsers, 16

TextPad,36

`<tfoot>` element, 144,359

`<th>` element, 144,359

`<thead>` element, 144,359

thumbnail images, 110,382

tiling images, 121

`title` attribute
`<applet>` element, 352
`<style>` element, 358

`<title>` element, 54,359

titles,382
page titles, 53–54,308

toolbox resources, 347

`<tr>` element, 360
`align` attribute, 151
attributes,360
hierarchy among other table elements, 140–141
`valign` attribute, 151–152

training resources, 346

transferring files
via FTP, 300–301
via hosting provider's Web site, 301

transparent images, 112

`<tt>` element, 128,360

TUCOWS Web site, 111,274,293

`type` attribute
`<a>` element, 352
`<button>` element, 353
`<input>` element, 187,355
`<link>` element, 356

`<object>` element, 357
`<ol>` element, 74
`<param>` element, 357
`<script>` element, 358
`<style>` element, 358
`<ul>` element, 75–76

Index

U

<u> element, 128,360

UI.*See*user interface (UI)

 element, 74-76,360

under construction messages, 314

underline
- hyperlinks,120
- text,128

Uniform Resource Locator. *See*URL

Unix,382

unordered lists, 74-76

updating Web sites, 301-305

URL,382
- bookmark,375
- broken links, 84-85
- case sensitivity, 84
- domain,19
- filename,19,85
- filename extension, 84
- how it works, 18-19
- http://,85
- hyperlinks,81-84
- path,19
- protocol,19

URL-encoded text, 383

usability
- forms,198
- frames,163
- resources available, 344

usemap attribute
- <img> element, 109,355
- <input> element, 355
- <object> element, 357

user feedback, 331-332

user interface (UI), 383
- accessibility,321
- design resources, 324
- examples,321-324
- goals of, 310-311
- multimedia,320
- navigation,309
- navigation schemes, 315-318
- off-site links, 318-319
- purpose of, 309
- site maps, 311-315

Index

V

valid,383

validating
- Cascading Style Sheets (CSS), 209
- forms,229-230
- HTML,286-289,331,337

validation,383

`valign` attribute
- `<col>` element, 353
- `<colgroup>` element, 353
- `<tbody>` element, 359
- `<td>` element, 151-152,359
- `<tfoot>` element, 359
- `<th>` element, 359
- `<thead>` element, 359
- `<tr>` element, 151-152,360

`value` attribute
- `<button>` element, 353
- `<input>` element, 355
- `<param>` element, 357

`valuetype` attribute, 357

`<var>` element, 129,360

VBScript,224

VeriSign Web site, 298

video
- embedding files into Web pages, 239-241
- links,237-239
- players,234-235
- streaming video, 242-243
- video formats, 234-235

viewing Web pages, 42-43

VIVO video format, 235

`vlink` attribute, 119,353

`vspace` attribute
- `<applet>` element, 352
- `<img>` element, 106-107

Index

W

WAI (Web Accessibility Initiative), 102

WAV (RIFF WAVE) audio format, 234

Web,383

Web browsers, 375
- alternative text, 100-102
- Cascading Style Sheets (CSS), 159,207-208
- character set support, 361
- client-side scripts, 223
- color settings, 119
- colors,116
- display instructions, 12-13
- frames,161,166
- graphical Web browsers, 16
- Hypertext Transfer Protocol (HTTP), 17
- image display, 100
- interpretation of HTML, 15-16
- line breaks, 67
- Lynx,16,379
- opening a new window, 86-87
- page title display, 54
- plug-ins,236
- redirecting users, 56-57
- scripting languages, 224
- streaming media, 242-243
- style sheets, 203,207-208
- testing,16,337,339
- text-only Web browsers, 16
- viewing Web pages, 42-43
- white space, 64-65
- XHTML,260,263
- XML (Extensible Markup Language), 260

Web Design For Dummies, Lisa Lopuck, 114,324

Web Design Group Web site, 212, 345

Web Developer's Journal Web site, 347

Web Developer's Library Web site, 345

Web Hosting Ratings Web site, 297

Web pages, 383
- advanced features, 33-34
- content types, 11-12
- creating from scratch, 36-43
- design tips, 338
- editing,43-45
- functionality,31-32
- hosting,291-297
- hosting services, 14-15
- `.htm` suffix, 42
- `.html` suffix, 42
- layout,337-338

linear versus nonlinear approach to design, 340-341
look and feel, 32
multimedia,33
naming,41-42
page titles, 308
planning for an entire Web site, 34
posting,45-47
publishing,45-47
saving,41
scripting,33
scripts,221
shopping cart, 33
testing on browsers, 16
viewing,42-43

Web Pages That Suck Web site, 324

Web Reference Web site, 346

Web server, 383
database connections, 256
dedicated Internet connections, 293-294
Hypertext Transfer Protocol (HTTP), 17
requirements,293
role of, 14-15
server-side scripts, 223

Web site resources
Adobe GoLive, 282
Adobe Photoshop, 283
Amazon.com,322-323
Apple QuickTime, 237
ASP.NET,256
BBEdit,280
Bobby,321
Bravenet, 347
Builder.com,96,257
CGI Resource Index, 197
DevGuru,206
Fetch,289
FirstGov,17-18
HomeSite,278
HTML Goodies, 348
HTML Writers Guild, 346
HTMLhelp, 345
Impact Online, 28
IRS,322,324
Jakob Nielsen's Useit, 324, 344
Jasc PaintShop Pro, 283
JavaScript City, 225
JavaScript Kit, 225
JavaScript Source, 225
Java-Scripts.Net,225
JSP,256
Kira's Web Toolbox, 347
Macromedia,235,277,283
Matt's Script archive, 197
Microsoft Access, 251
Microsoft FrontPage, 278

Microsoft SQL Server, 252
MOMSpider,285
MySQL,250
Oracle,252
PDImages.com,114
PHP,254
RateGenius,199
ScriptSearch,197
Sybase,252
TUCOWS,111,274,293
VeriSign,298
Web Design Group, 212
Web Developer's Journal, 347
Web Developer's Library, 345
Web Hosting Ratings, 297
Web Pages That Suck, 324
Web Reference, 346
Web Standards Project, 157
WebAttack, 347
Webmonkey,96,98,110,112,186,196,257,324, 347
World Wide Mart, 299
WS_FTP Pro for Windows, 289
W3C (World Wide Web Consortium), 29,261
W3C Link Checker, 284

Web sites, 383
accessibility,321
design tips, 338
expanding,305-306,339-340
ghost sites, 341
goals,310
hosting,291-297
layout,337-338
linear versus nonlinear approach to design, 340-341
look and feel, 32,311
maintaining,301-305,341
Mega Web Tools, 347
page redirectors, 330
promoting,307-308
scope,310
site maps, 311-315,327-328,336
site reviews, 332
transferring files, 300-301
'under construction' messages, 314
updating,301-305
user feedback, 331-332

Web Standards Project, 157

Web Usability For Dummies, Richard Mander and Bud Smith, 324

WebAttack Web site, 347

Web-hosting provider, 383

Webmonkey Web site
ASP resources, 257
CGI scripts article, 196
database resources, 257
Design Basics article, 324

GIF animation tutorial, 112
Good Forms article, 186
graphic formatting tutorial, 96
image map resources, 110
image optimization tutorial, 98
PHP resources, 257
resources available, 347
Site Redesign Tutorial, 324

Web-server software, 293

well-formed document, 383

What You See Is What You Get (WYSIWYG),383

white space, 64-65,148,383

`width` attribute
`<applet>` element, 352
`<col>` element, 353
`<colgroup>` element, 353
`<hr>` element, 69
`<iframe>` element, 355
`<img>` element, 102-103,355
`<object>` element, 357
`<pre>` element, 357
`<table>` element, 146
`<td>` element, 146

WinAmp MP3 player, 236

windows
opening a new window, 86-87
scripting,227-229

Windows Media Audio (WMA), 234

Windows Media Player, 237,241-242

Word documents, saving as text files, 38

Word97/2000 for Windows (Microsoft), 281

word processing programs, 36

WordPerfect 8.0 for Windows (Corel), 281

World Wide Mart Web site, 299

World Wide Web Consortium (W3C)
Cascading Style Sheets (CSS), 205
HTML 4.01 specification, 26
resources available, 343-344
standards,3
Web Accessibility Guidelines, 102,321
Web site, 29
XML,261

wrapping text, 67-68

WS_FTP Pro for Windows, 289-290,300

W3C.*See*World Wide Web Consortium (W3C)

WYSIWYG (What You See Is What You Get), 383

Index

X

XHTML,384
- browser support, 260,263
- differences from HTML, 260
- differences from XML, 260
- how it works, 263
- HTML Tidy, 268
- namespaces,267-268
- rules,265-268
- switching from HTML to XHTML, 264-265

XML (Extensible Markup Language), 384
- browser support, 260
- differences from HTML, 260
- differences from XHTML, 260
- DOCTYPE declaration, 267
- Document Type Definition (DTD), 262-263
- FAQ Web site, 262
- how it works, 261-262
- namespaces,267
- rules,265-268
- uses,261

XML For Dummies, 3rd Edition, Ed Tittel and Natanya Pitts, 263

Cheat Sheet: HTML 4 For Dummies, 4th Edition

Overview

Golden Rules of HTML

- Always nest correctly.
- Always include ending tags in tag pair elements.
- Always quote attribute values.
- Always begin HTML documents with a DOCTYPE declaration.
- Always include the `<html>`,`<head>`,`<title>`, and `<body>` elements in your document.

Layout and Structure Elements

Element	Name	Function
<!DOCTYPE>	Document type	Specifies the version of HTML used in the document
<address> - </address>	Attribution information	Lists author contact information
<body> - </body>	Body	Defines or indicates a document's body
-	Logical divisions	Marks divisions in a document
- -	Headings	Identifies first-level through sixth-level - headings
<head> - </head>	Head	Indicates a document's head
<html> - </html>	HTML document	Identifies an HTML document
<meta>	Meta-information	Describes aspects of the page's information structure, contents, or relationships to other documents
-	Localized style formatting	Applies style to subparts of a paragraph
<title> - </title>	Document title	Briefly describes document information
<!- - ->	Comments	Inserts comments are not displayed by browsers

Layout Elements

Element	Name	Function
<dd> - </dd>	Definition description	Marks the definition for a term in a glossary list
<dir> - </dir>	Directory list	Marks an unbulleted list of short elements
<dl> - </dl>	Definition list	Marks a special format for terms and their definitions
<dt> - </dt>	Definition term	Marks the term being defined in a glossary list
-	List item	Marks a member item within a list of any type
<menu> - </menu>	Menu list	Marks a pickable list of elements
-	Ordered list	Marks a numbered list of elements
-	Unordered list	Marks a bulleted list of elements

Text Layout Elements

Element	Name	Function
<abbr> - </abbr>	Abbreviation	Identifies expansion for an acronym
<acronym> - </acronym>	Acronym	Indicates an acronym
<blockquote>	Quote style	Sets off long quotations or citations -</blockquote>
 	Force line break	Forces a line break in the on-screen text flow
<cite> - </cite>	Citation markup	Marks distinctive text for citations
<code> - </code>	Program code text	Used for code samples
<del> - </del>	Deleted text	Identifies sections of a Web page deleted in revision
<dfn> - </dfn>	Defined term	Emphasizes a term about to be defined in the text
-	Emphasis	Emphasizes enclosed text
<ins> - </ins>	Inserted text	Identifies Web page sections inserted in revision
<kbd> - </kbd>	Keyboard text	Marks text entered by the user at the keyboard
-	Paragraph	Breaks text into content blocks
<pre> - </pre>	Preformatted text	Keeps spacing and layout of original text in monospaced font
<q> - </q>	Quotation markup	Marks a short quotation within a sentence
<samp> - </samp>	Sample output	Indicates sample output from a program or script
-	Strong emphasis	Provides maximum emphasis to enclosed text
₋	Subscript	Renders text smaller and slightly lowered
⁻	Superscript	Renders text smaller and slightly raised
<var> - </var>	Variable text	Marks variable or substitution for some other value

Wiley, the Wiley Publishing logo, For Dummies, the Dummies Man logo, the For Dummies BestsellingBook Series logo and all related trade dress are trademarks or registered trademarks of Wiley Publishing, Inc. All other trademarks are property of their respective owners.

Copyright © 2003 Wiley Publishing, Inc. All rights reserved.

Item 1995-6.

For more information about Wiley Publishing, call 1-800-762-2974.

List of Figures

Chapter 1: The Least You Need to Know about HTML and the Web

Figure 1-1:This Web page has several different components.

Figure 1-2:FirstGov is a gateway that uses hyperlinks to help visitors find government information on the Web.

Figure 1-3:The components of a URL help it define the exact location of a single file on the Web.

Figure 1-4:ASCII test can't represent all text characters so HTML entities do instead.

Figure 1-5:Always use entities when you want to display a less-than sign, greater-than sign, or ampersand in the browser window.

Chapter 2: HTML at Work on the Web

Figure 2-1:The Impact Online Web page is simple but direct - the best way to present its creator's message.

Figure 2-2:The W3C home page is complex and robust; it helps visitors access the hundreds of other pages on the W3C site.

Chapter 3: Creating Your First HTML Page

Figure 3-1:Taking a few minutes to sketch your page design makes writing HTML easier.

Figure 3-2:The letter that is the text for our page in word-processing form.

Figure 3-3:Choose an easy-to-access location and a descriptive filename for your HTML pages.

Figure 3-4:Use Internet Explorer to navigate to your Web pages.

Figure 3-5:Viewing a file in your Web browser.

Figure 3-6:Viewing an HTML file in your text editor and Web browser at the same time.

Figure 3-7:A change in the HTML is displayed in a browser after a quick save and refresh.

Figure 3-8:Drag and drop HTML files from your local system to a Web server to make them available on the Internet.

Figure 3-9:After you transfer a file to a Web server, it's available via a URL to anyone with a Web browser and live Internet connection.

Chapter 4: Structuring Your HTML Documents

Figure 4-1:HTML page titles usually appear in a Web browser's window title bar.

Figure 4-2:When you use a **<meta>** element to create a page redirector, include text that visitors can click if their browsers can't handle the redirector.

Figure 4-3:Only Content in the <body>element appears in the browser's window.

Figure 4-4:Web browsers delineate paragraphs with a line break and white space.

Figure 4-5:Use the **align** attribute with a paragraph to specify its horizontal alignment.

Figure 4-6:Web browsers display headings in decreasing size from level one to level six.

Figure 4-7: Web browsers typically indent a block quote to separate it from paragraphs.

Figure 4-8:Web browsers routinely ignore white space.

Figure 4-9: Use preformat-ted text to force browsers to recognize white space.

Figure 4-10: Use **<pre>** with **<block quote>** to control white space within a quotation.

Figure 4-11: Using the **
** element to specify where lines in block elements should break.

Figure 4-12:Use the **<hr>** element to add horizontal lines to your page.

Figure 4-13:Use the **<hr>** attributes to better control how a browser displays the rule.

Figure 4-14:The LANWrights, Inc. Web site uses hard rules to draw your attention to important information on the page.

Figure 4-15:Use the <ol> and <li> attributes to create a numbered list.

Figure 4-16: Web browsers set the numbers for your list according to the order items appear in the list.

Figure 4-17: Use the **start** and **type** attributes to guide the display of a numbered list in a browser.

Figure 4-18:An unordered list uses bullets instead of numbers to mark items.

Figure 4-19: Use the **type** attribute to change the bullet style for an unordered list.

Figure 4-20:Definition lists group terms and their related definitions into a single list.

Figure 4-21: Nested lists combine two or more lists for a multi-level organization of information.

Chapter 5: Linking to Online Resources

Figure 5-1:A paragraph with a link to the W3C.

Figure 5-2:Use the **target** attribute to open a new window in a Web browser to display a linked file.

Figure 5-3:Use anchor elements to mark spots on a page and link to them.

Figure 5-4:This browser prompts you to save or view the ZIP file.

Chapter 6: Finding and Using Images

Figure 6-1:The White House Web page uses images in a variety of ways.

Figure 6-2:Use the **<img>** element to place graphics in a Web page.

Figure 6-3:When a browser doesn't show an image, it shows alternative text instead.

Figure 6-4:Even when a browser shows an image, it may display the alternative text as a pop-up tip as well.

Figure 6-5:A small box.

Figure 6-6:A small box becomes a long line.

Figure 6-7:Don't use a browser to resize complex images.

Figure 6-8:Use the **`border`** attribute to create a border around your image.

Figure 6-9:You can vary image alignment to control image placement on the page.

Figure 6-10:The**`hspace`** and **`vspace`** attributes control the white space around an image.

Figure 6-11:Combine image and anchor elements to create a linked image.

Figure 6-12:Image maps turn different areas of an image into linking regions.

Chapter 7: Top Off Your Page with Formatting

Figure 7-1: Use the **`<body>`** element attributes to define the colors on your Web page.

Figure 7-2:A back-ground image can be any size; the browser tiles it to fill in the entire page background.

Figure 7-3:A Web page with both colors and a background image defined.

Figure 7-4:Be sure to coordinate your text colors and background colors, or your page may be unreadable.

Figure 7-5:Use the **`<font>`** element and the **`face`** attribute to change the font face for sections of text on the page.

Figure 7-6:Use the **`<font>`** element and the **`color`** attribute to change the font color for sections of text on the page.

Figure 7-7:Use the `<font>` element and the `size` attribute to change the font size for sections of text on the page.

Figure 7-8:Relative font sizes use positive and negative numbers to set font size relative to the browser's default font size.

Figure 7-9:You can often get the same results using relative font sizes or absolute font sizes.

Figure 7-10:Many of the font style and phrase elements display the same way in a Web browser.

Chapter 8: HTML Tables

Figure 8-1:This Web page uses three different tables for layout. Each table is numbered.

Figure 8-2:This Web page uses one simple table with three cells for its layout.

Figure 8-3:After you delete a few cell walls, an HTML table might not look like much of a table at all, but it's perfect for laying out Web page elements.

Figure 8-4:Always start by sketching the table dimensions, even before opening text editor.

Figure 8-5:The beginning og the table structure contains only two rows.

Figure 8-6:Most Web pages that use tables for layout don't use borders; here you can see why.

Figure 8-7:This image doesn't define width properties.

Figure 8-8:This image defines width properties.

Figure 8-9:Cellpadding increases the space within each cell.

Figure 8-10:Cellspacing increases the width of the border.

Figure 8-11:Our simple table centered.

Figure 8-12:The cell spans two columns.

Figure 8-13:We have changed our table design so that the last cell containing our navigational items spans two rows.

Figure 8-14:Our simple table with the last rows spanned.

Figure 8-15: Nested tables.

Chapter 9: HTML Frames

Figure 9-1:This site uses two frames.

Figure 9-2:A sketch of Figure 9-1.

Figure 9-3:Navigational elements are defined as a row on top of the content.

Figure 9-4:This Web page consists of two columns.

Figure 9-5:The Web page consists of two rows.

Figure 9-6:The Web page consists of three rows, with the third row taking up the remaining height in the browser.

Figure 9-7:This frame structure has three rows and two columns, which results in six separate frames.

Figure 9-8:Borders turned off using **`frameborder="0"`**.

Figure 9-9:Borders turned on using `frame-border="1".`

Figure 9-10: The browser window displays a scroll bar for the first frame, yet leaves it off for the second frame, even though the content is cut off.

Figure 9-11:The first frame, split into two smaller frames that contain navigational elements and a note to users.

Chapter 10: HTML Forms

Figure 10-1:The IRS home page uses two short search forms to help users find what they are looking for in minimum time.

Figure 10-2:The refund status search form is a little more complex.

Figure 10-3:A subscription form collects information to help teachers subscribe to an online newsletter.

Figure 10-4:An online car loan site uses many long and detailed forms to collect necessary data.

Figure 10-5:Text entry fields in a form.

Figure 10-6:You can specify the length and the maximum number of characters for a text field.

Figure 10-7:Password fields are like text fields except that the browser masks the text a user enters.

Figure 10-8: Check boxes and radio buttons.

Figure 10-9:A file upload field.

Figure 10-10:A drop-down list.

Figure 10-11: A drop-down list with modifications.

Figure 10-12:A text box.

Figure 10-13:Submit and reset buttons.

Figure 10-14:A form gateway page helps users prepare to fill out a long form.

Chapter 11: Getting Stylish with CSS

Figure 11-1:An HTML page without style specifications.

Figure 11-2:An HTML page with style specifications in effect.

Figure 11-3:Use classes to target your style rules more precisely.

Figure 11-4: You can create style rules that work with any element by using classes.

Figure 11-5:Inheritance means style rules apply to nested elements.

Chapter 12: HTML and Scripting

Figure 12-1:. . . a script makes a dialog box appear to tell you what you did wrong.

Figure 12-2:When the page loads, it displays the image referenced in the image **`src`** attribute.

Figure 12-3:When a mouse pointer moves over the image, the browser displays a different graphic.

Figure 12-4:You can use JavaScript to open a Web page in a new window and carefully control the display of the window.

Figure 12-5:A good use of JavaScript is to validate form data.

Chapter 13: Making Multimedia Magic

Figure 13-1:This site shows a link for downloading an MP3 file.

Figure 13-2:A console (defined as a control) is visible in the Web page.

Chapter 14: Integrating a Database into Your HTML

Figure 14-1: The components of a data-driven Web solution may exist on separate systems.

Chapter 16: Creating an HTML Toolbox

Figure 16-1:The New Document dialog box.

Figure 16-2:Adding a title to your HTML document.

Figure 16-3: HomeSite's interface displaying a blank HTML document.

Figure 16-4:Using the W3C Link Checker we can verify that all links found at www.lanw.com/staff are not broken.

Figure 16-5:The W3C Validator Web page is simple and easy to use.

Figure 16-6:Error message that found a missing closing **`</title>`** tag.

Figure 16-7: The basic WS_FTP Pro interface.

Chapter 18: Creating a Great User Interface

Figure 18-1: The site map for the *XML For Dummies,* 3rd Edition Web site.

Figure 18-2: The *XML For Dummies* site without the book examples.

Figure 18-3: The *XML For Dummies* site with the book examples.

Figure 18-4: The Dummies.com site is organized by topic.

Figure 18-5: The main topic areas on the Dummies.com site are accessible from the top navigation bar.

Figure 18-6: Main areas on the IRS site are accessible from a persistent navigation bar.

Figure 18-7: Each area of the site uses the same general layout.

List of Tables

Chapter 2: HTML at Work on the Web

Table 2-1: Adding Advanced Features to a Page

Chapter 6: Finding and Using Images

Table 6-1: Choosing the Right File Format

Chapter 7: Top Off Your Page with Formatting

Table 7-1: The 16 Color Names with Their Hex Codes

Appendix A: HTML 4 Tags

Table A-1: HTML Tags

Appendix B: HTML Character Codes

Table B-1: The ISO-Latin-1 Character Set

Table B-2: Greek Characters

Table B-3: Special Punctuation

Table B-4: Letter-like Characters

Table B-5: Arrow Characters

Table B-6: Mathematical Characters

Table B-7: Technical Characters

Table B-8: Playing Card Symbols

Table B-9: ISO 8859 Character Sets

List of Listings

Chapter 1: The Least You Need to Know about HTML and the Web

Listing 1-1: Sample HTML Markup

Chapter 3: Creating Your First HTML Page

Listing 3-1: The Complete HTML Page for the Zog Letter

Chapter 5: Linking to Online Resources

Listing 5-1: Intradocument Hyperlinks

Chapter 7: Top Off Your Page with Formatting

Listing 7-1: Using <body> Element Attributes to Control a Page's Colors

Listing 7-2: Text Treatments at Work

Chapter 10: HTML Forms

Listing 10-1: A Simple Form Processed by a Form Handler

Chapter 11: Getting Stylish with CSS

Listing 11-1: Adding an Internal Style Sheet to an HTML Document

Chapter 12: HTML and Scripting

Listing 12-1: Verifying the User Fills Out Both Form Fields

Listing 12-2: Image Rollover

Listing 12-3: Pop-Up Window

Listing 12-4: Form Validation

List of Sidebars

Chapter 1: The Least You Need to Know about HTML and the Web

A bevy of browsers

Technical Stuff Introducing Internet protocols

Chapter 5: Linking to Online Resources

Anchor elements aren't block elements

The importance of http:// in HTML links

Chapter 6: Finding and Using Images

Creating thumbnail images

Remember Copyright matters

Chapter 7: Top Off Your Page with Formatting

Remember Just because you can do something . . .

Chapter 8: HTML Tables

Other table elements

Chapter 11: Getting Stylish with CSS

Tip Paying attention to inheritance

Chapter 12: HTML and Scripting

Tip Finding Scripts Online

Chapter 14: Integrating a Database into Your HTML

Technical Stuff A bit about SQL

Tricks of the Trade The importance of good database design

Technical Stuff And now for some superfluous, geeky information

Tip Do it yourself or hire an expert?

Chapter 15: How HTML Relates to Other Markup Languages

Warning Avoiding mutant markup

Tip Keeping markup clean with HTML Tidy

Chapter 16: Creating an HTML Toolbox

Tip Where to go for more information

Tricks of the Trade The one-two punch of using a WYSIWYG editor with a helper editor

Warning Word processors and HTML

Chapter 18: Creating a Great User Interface

Tricks of the Trade Design matters