We could start this book with lots of exercises, throwing you right into working with Dreamweaver MX without any preparation. But then you would be flying blind, without understanding basic Web-design fundamentals such as HTML, DHTML, XML, XHTML, and JavaScript. Instead, we are starting you off with some definitions, concepts, and guidelines to help with your hands-on Dreamweaver MX training. Feel free to scan this chapter for information if you already know some of what is here or you want the instant gratification of getting started.
What Is Dreamweaver MX?

At its roots, Macromedia Dreamweaver MX is a WYSIWYG (What You See Is What You Get) HTML generator. This means if you change something on the screen inside Dreamweaver MX, it will show you the results instantly. In contrast, if you were to code the HTML by hand, you would have to look at the code until you viewed the results from inside a Web browser. The instant feedback of a live design environment such as Dreamweaver MX speeds up your workflow tremendously because you can see whether you like the results while you are working. However, if you really look under the hood of Dreamweaver MX, you will see that it is also a complete Web application development tool, capable of developing advanced Web applications, e-commerce sites, dynamic data-driven sites, and much more.

Roundtrip HTML

Dreamweaver MX has gained a lot of great reviews and customer loyalty because of its invention of Roundtrip HTML. Roundtrip HTML means you can easily move between Dreamweaver MX and another HTML text editor, such as FrontPage, BBEdit, or HomeSite, with very little or no change to your code. Unless you are a programmer (and chances are you aren’t if you are reading this book ;-) ), this probably won’t mean a whole lot to you right now. However, being able to move between different HTML development tools can be very important when you are working with a programmer or in a team environment where everyone might not be using Dreamweaver MX. It’s nice to know that you can do this and not worry about Dreamweaver MX breaking your code by inserting unwanted changes. Don’t you wish all programs were so respectful?

Programmers have looked at WYSIWYG HTML editors with dubious eyes because of their reputation for inflexibility and inclusion of nonstandard HTML code. Dreamweaver MX is one of the few WYSIWYG HTML editors to win the approval of programmers and designers alike. Programmers like the product because they don’t have to worry about their code being changed by Dreamweaver MX. Designers like Dreamweaver MX because it writes clean code without inserting a lot of proprietary and self-serving tags, and because it allows them to do visual layout without understanding a line of code. Hard to believe there could be a tool that could please both of these divergent groups, but there is, and Dreamweaver MX is it!

Now, truth be told, by default Dreamweaver MX does make some minor changes to a page once it’s opened. Because the few changes it makes are really aimed at cleaning up bad code, no one really frowns on these changes. There is no reason to get into those issues now; you’ll learn about these changes and how to turn them off (if you even want to do so) in Chapter 12, "HTML."
**Do You Need to Learn HTML to Use Dreamweaver MX?**

Yes and no. If you use a WYSIWYG HTML editor, then technically you can create an entire Web page without writing a single line of code. However, there may come a time when you will have to edit the code manually to troubleshoot a problem, such as an incompatibility between browsers. For some of us, HTML is quite intimidating at first glance—your first reaction may be to avoid it at all costs. After all, when you design pages using Photoshop, QuarkXPress, or PageMaker, you don’t need to look at raw PostScript code anymore. However, the early pioneers of desktop publishing had to know how to program in PostScript just to create a page layout! Most of the early Web developers were programmers, not artists, and they needed to write the raw code to create a Web page.

In the past, if you didn’t know some HTML, you were at the mercy of a programmer who might have had more control over your design than you liked. Today, with Dreamweaver MX, you can get by without understanding or writing a single line of code. However, if you long to be a true professional Web designer or developer, we strongly recommend that you take the time to learn HTML. We have found that most people who don’t learn a little HTML are at a disadvantage in the workplace, especially when they need to troubleshoot problems on their Web pages.

How do you learn HTML? You can take a class at school, take an online class, buy a training CD-ROM (we happen to know of a really good one!), or buy a book. There are endless ways to learn HTML; you should learn the way that works best for you. An easy way to learn HTML is to view the source code of pages that you like. In HTML the code is visible to everyone who uses a Web browser. To view the source code of a page, look under your browser’s *Edit* menu and choose *View > Page Source* (Netscape) or *View > Source* (Explorer). This will show you the raw HTML, and once you get comfortable with some of the tags, you will likely be able to deconstruct how these pages were made. And if all else fails, you can hire the neighborhood 12-year-old, who learned it last weekend instead of mowing your lawn. ;)

---

**What Is XHTML?**

HTML 4.01 is the most current version of HTML. As of October 4, 2001, the Web standards committee decided to discontinue HTML. Taking its place is **XHTML**, a language almost identical to HTML with the exception that it is written with different rules. Does this mean that if you learned HTML you wasted your time? Heck no. HTML will still be used, as well as XHTML, to create Web pages for a long time. In fact, knowing HTML will help you make the transition to XHTML, so don’t worry. You’ll find more details on XHTML later in this chapter in the “What About XHTML?” section.
HTML Resources

Many great resources are available, online and off, for learning HTML. Here are some online sites that are worth checking out.

World Wide Web Consortium
http://www.w3.org/MarkUp/

HTML: An Interactive Tutorial for Beginners

Webmonkey: HTML Tutorial
http://www.hotwired.lycos.com/webmonkey/teachingtool/index.html

NCDesign: HTML Design Guide v5.0
http://www.ncdesign.org/html/

Index DOT HTML: The Advanced HTML Reference
http://home.webmonster.net/mirrors/bloo-html/

The HTML Writers Guild: A Resource List
http://www.hwg.org/resources

What Does HTML Do?

HTML stands for HyperText Markup Language. It is a derivative of SGML (Standard Generalized Markup Language), an international standard for representing text in an electronic form that can be used for exchanging documents in an independent manner.

Back in 1980, people had to use a form of markup in word processor documents. If you wanted something to have a bold face, for example, you had to tag it with the symbol <B> in order to create that formatting. You would never see the actual boldfaced text until the file was printed; back then, bold type could not even be displayed on the computer screen!

We've come a long way since then, and so has HTML. That's why programs such as Dreamweaver MX have become viable alternatives to writing all the tags by hand. With maturity and established standards, HTML in its raw form will likely become as hidden as the markup behind word processors is today.
At its heart, HTML allows for the markup of text and the inclusion of images, as well as the capability to link documents together. Hyperlinks, which are at the core of HTML's success, are what allow you to flip between pages in a site, or to view pages in outside sites. These hyperlinks are references that are contained within the markup. If the source of the link moves, or the reference to the link is misspelled, it won’t work. One of the great attributes of Dreamweaver MX is that it has site-management capabilities, which will help you manage your internal links so they are automatically updated if the links are changed.

What Does HTML Look Like?

HTML uses a combination of tags, attributes, and values to generate its results. Here is a sample line of code that uses a tag, an attribute, and a value.

```
<BODY BGCOLOR="#FFCC33">

Tag        Attribute        Value
```

In this line of code, the tag is `BODY`, the attribute is `BGCOLOR`, and the value is `#FFCC33`. When put together, this collection of items within the brackets `< >` is called an element.

Many tags require opening and closing containers, as marked here for the `<BODY>` element.
Dreamweaver MX H•O•T | 1. Background

HTML Deconstructed

All HTML pages follow a basic structure. Each page must contain the HTML, HEAD, TITLE, and BODY tags. Whenever you open a new HTML document in Dreamweaver MX, this framework is already written. It is deconstructed for you below. Fortunately, you don’t have to worry about getting this structure right. It is automatically built in to any page you create in Dreamweaver MX.

1. <HTML>
2. <HEAD>
3. <TITLE>Untitled Document</TITLE>
4. <META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
5. </HEAD>
6. <BODY BGCOLOR="#FFFFFF">
7. </BODY>
8. </HTML>

1. Notice how the <HTML> tag is at the beginning of the document? It signifies that this is an HTML page. Without this tag, a browser cannot read the page. See line 8, the close </HTML> tag? This tag requires an open and a close tag. A closing tag is required for most HTML tags, but not all.

2. The <HEAD> element of the document contains the HEAD information, or the hidden information about your page. For example, the TITLE tag and the META tag are contained within the HEAD.

3. The <TITLE> is what appears at the top of the page inside a browser. If you leave the title Untitled Document, as in the previous code example, that is exactly what will appear! Dreamweaver MX has a setting for easily changing this title. You’ll learn about this setting in Chapter 4, “Basics.”

4. META tags are HEAD elements that are invisible when viewed in the browser, but contain information about the current page, such as the character encoding, author, copyright, and keywords. You can set many properties here, which you will learn about in Chapter 4, “Basics.”

5. Here’s the close tag for the HEAD element. Notice that the TITLE and META tags were nested within the HEAD tags.

6. The BODY tag is using the BGCOLOR attribute to specify that this page will be white, instead of the default gray. If you don’t enter a BGCOLOR value here, the page will defer to browser defaults.

7. This is the close tag for BODY.

8. This is the close tag for HTML.
File-Naming Conventions

Working with HTML is much more restrictive than working with other types of computer media. The strictest part about HTML is its file-naming conventions.

Don't use spaces: It's best if you save your files using no spaces between the file name elements. For example, the file name about lynda.html would be considered illegal because of the space between the words about and lynda. Instead, you would write this file name as about_lynda.html or aboutlynda.html.

Avoid capital letters: It is best to avoid capitalization in your file names. Although AboutLynda.html will work as a file name, anytime you link to the file you will have to remember the correct capitalization because many UNIX servers are case-sensitive. It is far easier to simply use all lowercase letters.

Avoid illegal characters: The following chart contains a list of characters to avoid when naming files.

<table>
<thead>
<tr>
<th>Character</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>.</td>
<td>Periods are reserved for file name extensions or suffixes, for example, .gif and .jpg.</td>
</tr>
<tr>
<td>&quot;</td>
<td>Quotes are reserved for HTML to indicate the value of tags and attributes.</td>
</tr>
<tr>
<td>/ or \</td>
<td>Forward slashes (/) indicate that the files are nested in folders. If you include a forward slash in your file name, HTML may lose your references, thinking you are specifying a folder. A backslash () isn't allowed on Windows servers.</td>
</tr>
<tr>
<td>:</td>
<td>Colons are used to separate certain script commands on Macs and Windows. Avoid them in your file names so as not to confuse a file name with a script command.</td>
</tr>
<tr>
<td>!</td>
<td>Exclamation marks are used in comment tags.</td>
</tr>
</tbody>
</table>
File Name Extensions

You may be curious about the many extensions used after the dot at the end of file names. The following chart lists the meaning of some extensions you'll commonly run across.

<table>
<thead>
<tr>
<th>File Name Extensions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Extension</strong></td>
</tr>
<tr>
<td>.html, .htm</td>
</tr>
<tr>
<td>.gif</td>
</tr>
<tr>
<td>.jpg</td>
</tr>
<tr>
<td>.swf</td>
</tr>
<tr>
<td>.mov</td>
</tr>
<tr>
<td>.avi</td>
</tr>
<tr>
<td>.aif</td>
</tr>
</tbody>
</table>
What About XHTML?

As you saw earlier, the current version of HTML is 4.01. There will not be a version 5 of HTML. Instead there will be XHTML 1.0. In fact, XHTML 1.0 already exists as a formal recommendation sanctioned by the World Wide Web Consortium (the standards committee of the Web). However, we are at a crossroad because XHTML is just now being accepted and supported by some software vendors, which means that HTML is still the prominent language used to create Web pages. The future of XHTML looks very promising, however, and we want you to be aware of its existence and intended purpose. The next version of XHTML, version 1.1, is already in development.

XHTML stands for eXtensible HyperText Markup Language. How is XHTML different from its close companion HTML? The most visible difference between the two markup languages can be seen in their syntax, with all opening tags requiring a closing tag. Here are some of the key differences:

• All element and attribute names are in lowercase. For example, `<P>` would not be valid, but `<p>` would be a valid XHTML element.

• All attribute values must be contained within quotes, single or double. For example, in HTML you can write `<td nowrap>`, but in XHTML you have to write `<td nowrap="nowrap">`. Make sure you are consistent in the type of quotes you use; don’t mix single quotes with double quotes, or vice versa.

• All nonempty elements must have a closing tag. For example, `<p>this is good text.</p>` is a valid XHTML element, whereas `<p>this is bad text.</p>` would not be valid.

• All empty tags should be written with a space and a `/` symbol at the end of the tag. For example, `<br />` is a valid XHTML tag; `<br>` is not. This method of closing empty tags ensures that your pages are compatible with older browsers.

XHTML follows the XML rules and syntax guidelines. Because XML has very rigid requirements for writing code, XHTML is a more structured markup language than HTML. This more structured approach to markup languages will enable one document to be viewed on multiple devices (Web browsers, cell phones, PDAs, etc.) by simply creating different style sheets for each device. (You will learn about style sheets later in the book.) In a nutshell, XHTML is basically HTML 4.01 reformatted using the syntax of XML. You will be glad to know that Dreamweaver MX has full support for XHTML. In fact, Dreamweaver MX can even convert your existing HTML documents to XHTML. You will learn more about this in Chapter 12, “HTML.” Sweet.
What Does XHTML Look Like?

If you have ever seen HTML code, you will find instant comfort in looking at XHTML code. Because XHTML is a reformating of HTML, many things look the same or have only minor differences. Although some distinct and critical differences exist between XHTML and HTML, they are both markup languages and share many common traits. This is good news because it will lessen the learning curve for those of you familiar with HTML.

Here are some of the basic elements of an XHTML document written in correct syntax.

1. `<?xml version="1.0"?>`
2. `<!DOCTYPE html public "-//W3C//DTD XHTML 1.0 Strict//EN"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd`
3. `<html xmlns="http://www.w3.org/1999/xhtml">
4. `<head>
5. `<title>
6. `</title>
7. `</head>
8. `<body>This is where the content of your page will be placed.
9. `</body>
10. `</html`

XHTML Code Deconstructed

1. The XML declaration. This line identifies this document as an XHTML document.

2. The Document Type Definition (DTD). This URL points to a file that outlines available elements, attributes, and their appropriate usage. Three XHTML DTDs are available:
   - XHTML Transitional: This DTD lets you maintain backwards compatibility with older browsers while still providing access to HTML 4.01 elements.
   - XHTML Strict: This DTD does not provide access to any of the HTML elements that were designed to control the appearance of a page. This is the truest form of XHTML elements.
   - XHTML Frameset: This DTD gives you access to the HTML elements needed to create framesets.

3. The XHTML Namespace. This URL points to a file that gives detailed information about the particular XML vocabulary, which is XHTML in this case.

4. The `<head>` tag contains all of the header information.
5. The `<title>` tag defines the page title, which appears at the top of the browser window and in the bookmark lists.

6. All XHTML tags must be closed, so this is the closing `<title>` tag.

7. You guessed it! This is the closing `<head>` tag.

8. All of your visible content will be placed inside the `<body>` tag.

9. Yup, here is another closing tag. This is the closing `<body>` tag.

10. Last, but not least, is the closing `<html>` tag.

This example represents only a smidgen of the available XHTML tags, attributes, and values. But it shows the basics and is a great place to start your XHTML education. We cover more examples of XHTML in Chapter 12, "HTML."

**What Is XML?**

XML stands for eXtensible Markup Language. XML is a set of guidelines for delimiting text through a system of tags so it can be read and processed by a device capable of reading a text file. You can think of it as a really customizable HTML that must follow a set of specific syntax rules. Because XML is a text format, you can imagine why so many developers like to work with XML data; you can do just about anything with a text file, regardless of what computer and operating system you are using. For this reason, XML is used to move data between different computers and different operating systems, which makes it perfect for e-commerce solutions and sending and retrieving data from a database.

Dreamweaver MX supports templates, covered in Chapter 17, "Templates and Library Items." One of the advanced features of Dreamweaver MX is the ability to export/import XML files through a template. Because XML is so complex and deep, and the use of databases is outside the scope of this book, we don’t include any XML exercises in any of the chapters. Here are some places you can go to learn more about XML:

**Macromedia Designer & Developer's XML Site**
http://www.macromedia.com/desdev/topics/xml.html

**World Wide Web Consortium**
http://www.w3.org/xml/
Extending Dreamweaver MX

One of the neatest things about the Dreamweaver MX community is the way people share objects, commands, and behaviors, which are like plug-ins for Dreamweaver MX that let you add programming functionality without typing a single line of code. These prebuilt elements can be shared and distributed, much the way Photoshop plug-ins work. If you visit the Macromedia Dreamweaver MX Exchange (http://exchange.macromedia.com/dreamweaver) section of the Macromedia site, you’ll find numerous ways to get more out of Dreamweaver MX without having to learn a complex programming language. In addition, you’ll find a collection of third-party sites that can help you extend the capabilities of Dreamweaver MX. Here are a few of our favorites:

**Project VII**
http://www.projectseven.com/
This site has a great collection of Design Packs and Extension Kits that can help you get a lot more from Dreamweaver MX. Some of them must be purchased for a fee.

**Dreamweaver Extensions Database**
http://www.idest.com/cgi-bin/database.cgi
Features an extensive database, which includes all of the Dreamweaver MX extensions.

**Yaromat**
http://www.yaromat.com/dw/index.php
A personal home page that contains several very useful Dreamweaver MX extensions, including a great one for importing Fireworks-created rollovers.

**Massimo’s Corner of the Web**
http://www.massimocorner.com/
A great resource for Dreamweaver MX extensions, Objects, Commands, and Behaviors. It has an interesting DHTML interface, too!

**Dreamweaver Designer & Developer Center**
http://macromedia.com/desdev/mx/dreamweaver/
A new part of Macromedia’s Web site that contains a large collection of articles, tips, and tutorials. If you are a Dreamweaver MX user, you should definitely visit this site.
What Is DHTML?

DHTML (Dynamic HTML) is a collection of different technologies. This can include any combination of HTML, JavaScript, CSS (Cascading Style Sheets), and DOM (Document Object Model). By combining these technologies, you can author more dynamic content than what basic HTML affords.

Some of the things possible with DHTML include animation, drag-and-drop, and complicated rollovers (buttons that change when a mouse moves over them). Dreamweaver MX uses DHTML to enable you to create pages with buttons that change in more than one place on the screen at the same time.

Just like HTML, DHTML effects in Dreamweaver MX are coded behind the scenes. However, DHTML has some serious cross-platform issues, because it is supported quite differently by Netscape and Explorer (the two leading browsers). Fortunately, Dreamweaver MX lets you target specific browsers, as well as test the cross-browser compatibility of your DHTML effects.

DHTML uses a combination of HTML, JavaScript, CSS, and DOM. The following chart gives a short description of each.

<table>
<thead>
<tr>
<th>DHTML Terms</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Technology</strong></td>
</tr>
<tr>
<td>HTML</td>
</tr>
<tr>
<td>JavaScript</td>
</tr>
<tr>
<td>CSS</td>
</tr>
<tr>
<td>DOM</td>
</tr>
</tbody>
</table>
What Is JavaScript?

JavaScript was developed by Netscape in 1996 and has become almost as popular as HTML. It actually has nothing to do with the Java programming language, but Netscape licensed the name from Sun Microsystems in hopes of increasing acceptance of the new scripting protocol. We’re not sure if it was the name that did the trick, but JavaScript has become almost as widely adopted as HTML itself! The most common uses of JavaScript allow for rollovers, resizing of browser windows, and checking for browser compatibility.

Most of the JavaScript routines are accessed by the Dreamweaver MX Behaviors panel, which you will learn about in Chapter 10, “Rollovers,” and Chapter 14, “Behaviors.” This book covers many JavaScript techniques, including rollovers, browser-sniffing, and launching external browser windows.

You will not have to learn to write JavaScript by hand in order to use it within Dreamweaver MX. This is very fortunate for those of us who are not programmers because JavaScript programming is more complicated than HTML programming. For those of you who are JavaScript programmers, however, Dreamweaver MX offers a JavaScript Debugging feature. This feature is outside the scope of this book, but you will find documentation on it in the Dreamweaver MX manual or online Help system.

What Is a Web Application?

In broad terms, a Web application is a Web site that delivers dynamic data, such as Amazon.com and eBay.com, instead of static data that has to be updated manually. Web applications have also been referred to as data-driven, database-driven, and dynamic sites. In almost all cases, a Web application involves a database and server-side scripting, such as ASP, Cold Fusion, PHP, etc. Web applications aren’t just one thing; they can take on many forms and serve many purposes. They can be used to handle e-commerce, inventory tracking, online auctions, and just about anything that uses a large amount of information. So what do Web applications have to do with learning Dreamweaver MX? Dreamweaver and UltraDev have been combined into one product, Dreamweaver MX. This means that Dreamweaver MX has the capability to create complete Web applications, in addition to static Web sites. Although creating Web applications is outside the scope of this book, you should know that you can use Dreamweaver MX to create them. As you advance your skills, you will find that you will not outgrow Dreamweaver MX, because the sky’s the limit as far as its capabilities are concerned.

Now that you have a basic foundation in these key areas, you are ready to learn more about Dreamweaver MX itself. The next chapter introduces you to the Dreamweaver MX interface and prepares you for the many step-by-step exercises throughout this book.