Chapter 3

DEVELOPING STATIC WEBSITES

What we’ll cover in this chapter:

- Site editing tools
- World Wide Web Consortium (W3C)
- XHTML
- Cascading Style Sheets
- JavaScript
- Templates and Library Items
- XHTML time
CHAPTER 3

Before you plunge into the waters of dynamic websites, you should learn the basics first. "Don't run before you can walk" is currently more applicable in the world of web development than it has ever been. Why? The way the web works is changing. Hacked up old HTML documents full of `<font>` tags are out; unforgiving XHTML and CSS documents are very much here to stay, and the Web is to be made accessible for everyone. In this chapter, you'll learn about why this is happening, what it means for you as a developer, and, more importantly, what on earth XHTML and CSS are actually all about. By the end of this chapter, you'll have made a basic web page that is totally XHTML/CSS valid and meets the new accessibility laws. It will have alternate style sheets, including large text and reverse colors for the visually impaired, making it easier for them to read.

Overview

For years, people have talked about making websites for a living or as a hobby. More often than not, it was very difficult to get these sites to work the same in the main two browsers (PC Internet Explorer and PC Netscape 4+). You would code some elements that would look fine in one browser, but in another it would look awful. Compromise was often the order of the day. The choices were

■ Use a detection script to send the user to a browser-specific version of the site (a very popular technique that's still in use today).

■ Design purely for PC Internet Explorer, which is currently the most popular browser in use (a bad technique, as it blocks some users, creating a bad impression).

■ Get the site looking "near enough" as you want it to in the main two browsers, and ignore all the other users. (Far too common. I'm sure some designers never knew there was life outside IE and Netscape.)

■ Code properly! (More about this later.)

In all of these scenarios, one set of users was guaranteed to come off worst of all: Mac users. I'd be willing to put all my Macs up as a stake and bet that nine out of ten designers who designed websites on a PC have never tested their work on a Mac. The sad fact is that the majority don't seem to care.

I, for one, always thought about both platforms, as I used both of them at work and home. I would often design a site at work using Windows as the test platform and the site would work just fine. Testing it at home on my Mac though was an entirely different story: There would be large gaps in the layout, misbehaving tables, and general funk occurring all over the place.

A designer could make far too many mistakes and simply get away with it. "Missed some closing tags off? Not a problem, I'll just render the page the way I think it should be!" said Internet Explorer, while Netscape would crank up the king of all tags `<blink>`, and actually flash where the error was! Of course, a lot of this bad coding was done by hand and usually to impossible deadlines for evil tyrant dictators (a.k.a. The Client). No syntax coloring + no visual guides = No Fun, and a greater risk of mistakes being made. I regularly used to code 90K HTML pages in Notepad that were full of nested tables. Eating in a restaurant one night after work, a waiter said to me:

"I'm afraid there's a problem with your table, sir."

"Check both the `<tr>` and `<td>` tags are closed properly," I said, without even thinking.

The look of confusion on the guy's face was classic, until I realized and explained. That's when I knew things really had to be done a different way. I managed to convince my boss to let us use Dreamweaver, and deadlines were never as scary from that day forth.

I started in this business by hand coding, and I'm still glad I did. I know how things work, and why. I can fix bits of code if I need to and make tiny adjustments without having to start up Dreamweaver. Hand coding is very useful, but let's be realistic here. If a client is banging on the door, wondering where their overdue 200-page site is, it's time for action.