Chapter 8

How Can I Add Interactivity?

Static websites are the bread and butter of web design. Being able to pull HTML and CSS together to produce an elegant but attractive website is essential to drawing visitors into the content you offer. But what if you want more than just a pretty face? What if you want to give your website and its visitors a more tailor-made design with interactive elements, special effects, and customized functionality to meet the needs of the visitors? Well, it is time to start enhancing the site’s behavior by looking at the various available scripting languages and packages, both client-side and server-side, that make your site react to the actions of your users. We’ll also go beyond simple scripting to take a look at plug-ins for providing rich interactivity and web applications, such as Flash and Silverlight. Scripting is tricky, and some tasks could be slightly beyond your skill level right now, but it’s not an essential addition to your website, so you can skip this chapter and come back to it later if you would like. There are some very cool things you can do to enhance your existing website, though, and languages like JavaScript can be of serious benefit to your website and how it functions. This chapter will give you a good solid idea about why scripting exists and how you can get started, but to learn more, you should check out a dedicated book, such as *Getting StartED with JavaScript* by Terry McNavage (ISBN-13: 978-1-4302-7219-9).

In this chapter, we’re going to cover the following topics:

- The basics of client-side scripting like JavaScript and using cookies
- Existing server-side packages and how you can use them
- Frameworks, scripting security issues, and rich Internet applications

Back to Basics

Here we go again! I expect the prospect of learning yet another language is rather daunting. Hopefully, at this stage, you have a website that works
perfectly fine, and while it may not have all of the pizzazz of some of the other sites you probably frequent, you should be proud of what you have accomplished. First, I think the good news should be given to you: the content within this chapter can enhance your website but is not actually a requirement! Many websites work perfectly fine with a mixture of HTML and CSS without a single line of script, and even if you do want to add some interactivity to your site, there are so many excellent premade packages and applications that you can plug into your site that you probably will not need to do any additional coding unless you really want to. You’ve already learned the compulsory languages. Of course, there are limits to what you can achieve without the use of scripting, and if you want to deviate at all from a prepackaged solution, you’re going to have to learn a little scripting, which can be a daunting task because scripting languages are complex. But never fear! I will start at the beginning and bring you up to speed with what’s out there.

Under the Hood

After all the hard work you’ve put in learning HTML and CSS, you should be well equipped to deal with the basics of scripting and seeing what goes on behind the scenes. Also, in the previous chapter on problem solving, you gained the rudimentary skills required to understand the process of isolating basic issues and working out where a mistake occurred. Unfortunately, with scripting, as you will soon find out, things are a little more technical, but so much work has been done in making things easier for the average user that the amount of work you need to do can be drastically reduced. But before we even come to looking at scripting code, you need to get to grips with the various languages involved. You can choose among lots of different scripting languages (unlike HTML and CSS where your choices are pretty standardized) especially at the server-side, and each has its own applications, tools, and communities to support people who make use of them, so let’s go back to basics!

Understanding Interactivity

Before I can describe what scripting is and how it works, I need to first discuss interactivity. Since the Web was created, the idea that content could be more than just text has been at the base of its evolution. While CSS (as you now know) gave website owners the opportunity to give their designs a greater level of detail than HTML alone could offer, the combination of the two languages never really gave websites the notable level of interactivity we see today, whether a website allows you to sign up for an account (and therefore remembers your details) or some exciting animated special effects go off all around your screen to make browsing around a more enjoyable experience.