

Trends

As website owners and those responsible for website success, it's imperative that we master the current trends in design and development. While you may be content to sit back and utilize the existing resources you have at your command and maximize the investment you have already made, it should be obvious that you are not in a static industry but rather a dynamic, thriving space where change is not just expected but usually embraced. And if you don't follow suit, you will become stagnant while the others around you catapult ahead.

A Look at the Trends

The problem in discussing trends in development is that trends, by their very nature, tend to become dated quickly — very quickly in some instances. But fortunately, programmers have taken this into account and the languages that were trends several years ago are still being used full force today. The only difference is that now, in many instances, we're able to build upon existing languages and create a more usable, accessible and interactive environment for users. In order to master the trends it is important to understand the components of what we will be mastering. The following is an overview of development languages and platforms that the Web is increasingly being built upon.

PHP: An open-source, reflective scripting language, PHP is popular with Web developers creating serverside application software. Most often used in tandem with MySQL databases, PHP features high-speed scripting with caching and is augmented with compiled code plug-ins. Learn more at PHP.net.

BLOCKED SCRIPT: JavaScript is a scripting language similar in syntax to the C programming language that runs in the user's Web browser. Thanks to a simplified set of commands this language is easier to create and implement. This is evident by its use in everything from opening new browser windows to offering link mouseover functionality.

AJAX: While not yet the defacto method for building Web applications, Ajax is getting close. It enables developers to create Web pages that retrieve data for users while they are busy interacting with other areas of the site — by merging data, content and design into a new interface. Wildly popular with many Web 2.0 sites, many feel it represents the next great iteration of the Web.

Ruby: A dynamic, object-oriented, open-source programming language, many consider Ruby a programmer's best friend. Usage has spread quickly thanks to its focus on simplicity, balance and productivity. Learn more at Ruby-Lang.org.

Perl: An open-source, cross-platform, server-side interpretive programming language used extensively to process text through CGI programs. Perl's power makes it easy to

write Web server programs for a range of tasks and is popular with those developing Web applications.

Mastering New Languages, Frameworks and Standards

Study every day: Learning anything of substance requires dedication. If you're serious about mastering new languages, frameworks and standards many experts agree that it's better to study for a short period of time every day than for several hours infrequently.

Reach out: There are thousands of excellent resources available on the Web to help you learn the aforementioned programming languages. But consistency is important in learning languages, such as those listed in the TOP50 section on page 15. Think of it not as a short term project but more of a long-term relationship.

Build on solid foundations: Make sure you're comfortable with the fundamentals of the language before tackling the more advanced stuff. It will take you a while to become familiar with the syntax of a new language, but without a thorough understanding you will find it impossible to create what these languages can offer.

Be patient, progress may be slow: While you may master some things quickly, be aware that there will be times that progress is slow or none at all. If you feel like little progress is being made, try reviewing earlier exercises and lessons to see if you missed something important.

You will make mistakes: Most of us make mistakes in our daily life, and making mistakes while learning new programming languages is no different. In fact, many agree that you can actually learn more by failing a few times — maybe even causing a few fatal errors. It's helpful, however, to have a guide in your programming studies. Ask that person to point out the more serious mistakes you have made and suggest how you can correct them. You will learn faster and develop a more refined degree of fluency.