The most popular programming languages for 2013 are the following:

1. Python
2. Java
3. C++
4. Ruby
5. JavaScript
6. C#
7. C
8. PHP
9. Perl

**Python** is a widely used [general-purpose](http://en.wikipedia.org/wiki/General-purpose_programming_language), [high-level programming language](http://en.wikipedia.org/wiki/High-level_programming_language).Its design philosophy emphasizes code [readability](http://en.wikipedia.org/wiki/Readability), and its syntax allows programmers to express concepts in fewer [lines of code](http://en.wikipedia.org/wiki/Lines_of_code) than would be possible in languages such as [C](http://en.wikipedia.org/wiki/C_%28programming_language%29). Python has distributions available for Microsoft Windows, Apple Mac OS X, GNU/Linux, BSD, web browser environments (NCLab, Sage) and many other platforms. There are currently three major implementations: the standard implementation written in C, Jython written in Java, and IronPython written in C# for the .NET environment. There are two common versions currently in use: 2.x and 3.x.

**Java** is a multi-platform language that is especially useful in networking. Of course, the most famous usage of Java is on the web, with Java applets, but Java is also used to build cross-platform programs that stand alone. Since it resembles C++ in syntax and structure, learning Java is usually quite easy for most C++ programmers. Java offers the advantages provided by object-oriented programming, such as reusability; on the other hand, it can be difficult to write highly efficient code in Java, and Swing, its primary user interface, is notoriously slow. Nevertheless, Java has increased in speed in recent years, and version 1.5 offers some new features for making programming easier.

**C++** is well-suited for large projects because it has an object-oriented structure. People can collaborate on one program by breaking it up into parts and having a small group or even one individual work on each part. The object-oriented structure also allows code to be reused a lot, which can cut down development time. C++ is also a fairly efficient language - although many C programmers will disagree.

**Ruby** is a [dynamic](http://www.wikipedia.org/wiki/Dynamic_programming_language), [reflective](http://www.wikipedia.org/wiki/Reflection_%28computer_science%29), general purpose [object-oriented programming language](http://www.wikipedia.org/wiki/Object-oriented_programming_language) that combines syntax inspired by [Perl](http://www.wikipedia.org/wiki/Perl) with [Smalltalk](http://www.wikipedia.org/wiki/Smalltalk)-like features. Ruby originated in [Japan](http://www.wikipedia.org/wiki/Japan) during the mid-1990s and was first developed and designed by [Yukihiro "Matz" Matsumoto](http://www.wikipedia.org/wiki/Yukihiro_Matsumoto). It was influenced primarily by [Perl](http://www.wikipedia.org/wiki/Perl), [Smalltalk](http://www.wikipedia.org/wiki/Smalltalk), [Eiffel](http://www.wikipedia.org/wiki/Eiffel_%28programming_language%29), and [Lisp](http://www.wikipedia.org/wiki/Lisp_%28programming_language%29). Ruby supports multiple [programming paradigms](http://www.wikipedia.org/wiki/Programming_paradigm), including [functional](http://www.wikipedia.org/wiki/Functional_programming), [object oriented](http://www.wikipedia.org/wiki/Object-oriented_programming), [imperative](http://www.wikipedia.org/wiki/Imperative_programming) and [reflective](http://www.wikipedia.org/wiki/Reflection_%28computer_science%29). It also has a [dynamic type](http://www.wikipedia.org/wiki/Dynamic_type) system and automatic [memory management](http://www.wikipedia.org/wiki/Memory_management); it is therefore similar in varying respects to [Python](http://www.wikipedia.org/wiki/Python_%28programming_language%29), [Perl](http://www.wikipedia.org/wiki/Perl), [Lisp](http://www.wikipedia.org/wiki/Lisp_%28programming_language%29), [Dylan](http://www.wikipedia.org/wiki/Dylan_%28programming_language%29), [Pike](http://www.wikipedia.org/wiki/Pike_%28programming_language%29), and [CLU](http://www.wikipedia.org/wiki/CLU_%28programming_language%29). The standard 1.8.7 [implementation](http://www.wikipedia.org/wiki/Ruby_MRI) is written in [C](http://www.wikipedia.org/wiki/C_%28programming_language%29), as a single-pass [interpreted language](http://www.wikipedia.org/wiki/Interpreted_language).

**JavaScript** (**JS**) is a [dynamic](http://en.wikipedia.org/wiki/Dynamic_programming_language) computer [programming language](http://en.wikipedia.org/wiki/Programming_language). It is most commonly used as part of [web browsers](http://en.wikipedia.org/wiki/Web_browser), whose implementations allow [client-side scripts](http://en.wikipedia.org/wiki/Client-side_scripting) to [interact with the user](http://en.wikipedia.org/wiki/User_interface), control the browser, communicate [asynchronously](http://en.wikipedia.org/wiki/Ajax_%28programming%29), and alter the [document content](http://en.wikipedia.org/wiki/Document_Object_Model) that is displayed. It is also being used in server-side programming, game development and the creation of desktop and mobile applications. JavaScript is a [prototype-based](http://en.wikipedia.org/wiki/Prototype-based_programming) [scripting language](http://en.wikipedia.org/wiki/Scripting_language) with [dynamic](http://en.wikipedia.org/wiki/Dynamic_language) typing and has [first-class functions](http://en.wikipedia.org/wiki/First-class_functions). Its [syntax](http://en.wikipedia.org/wiki/JavaScript_syntax) was influenced by [C](http://en.wikipedia.org/wiki/C_%28programming_language%29). JavaScript copies many names and naming conventions from [Java](http://en.wikipedia.org/wiki/Java_%28programming_language%29), but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the [Self](http://en.wikipedia.org/wiki/Self_%28programming_language%29) and [Scheme](http://en.wikipedia.org/wiki/Scheme_%28programming_language%29) programming languages. It is a [multi-paradigm](http://en.wikipedia.org/wiki/Multi-paradigm) language, supporting [object-oriented](http://en.wikipedia.org/wiki/Object-oriented_programming), [imperative](http://en.wikipedia.org/wiki/Imperative_programming), and [functional](http://en.wikipedia.org/wiki/Functional_programming) programming styles.

**C♯** (pronounced as *see sharp*) is a [multi-paradigm programming language](http://en.wikipedia.org/wiki/Multi-paradigm_programming_language) encompassing [strong typing](http://en.wikipedia.org/wiki/Strong_typing), [imperative](http://en.wikipedia.org/wiki/Imperative_programming), [declarative](http://en.wikipedia.org/wiki/Declarative_programming), [functional](http://en.wikipedia.org/wiki/Functional_programming), [procedural](http://en.wikipedia.org/wiki/Procedural_programming), [generic](http://en.wikipedia.org/wiki/Generic_programming), [object-oriented](http://en.wikipedia.org/wiki/Object-oriented_programming) ([class](http://en.wikipedia.org/wiki/Class_%28computer_science%29)-based), and [component-oriented](http://en.wikipedia.org/wiki/Component-based_software_engineering) programming disciplines. It was developed by [Microsoft](http://en.wikipedia.org/wiki/Microsoft) within its [.NET](http://en.wikipedia.org/wiki/.NET_Framework) initiative and later approved as a standard by [Ecma](http://en.wikipedia.org/wiki/Ecma_International) (ECMA-334) and [ISO](http://en.wikipedia.org/wiki/International_Organization_for_Standardization) (ISO/IEC 23270:2006). C♯ is one of the programming languages designed for the [Common Language Infrastructure](http://en.wikipedia.org/wiki/Common_Language_Infrastructure). C# is built on the syntax and semantics of C++, allowing C programmers to take advantage of .NET and the common language runtime. C♯ is intended to be a simple, modern, general-purpose, object-oriented programming language. Its development team is led by [Anders Hejlsberg](http://en.wikipedia.org/wiki/Anders_Hejlsberg). The most recent version is C♯ 5.0, which was released on August 15, 2012.

**C** is a popular language, especially in game programming, because it doesn't have the extra packaging of the object-oriented C++. Programmers use C because it makes programs slightly faster and smaller than programs written in C++. You might wonder, however, whether it's worth giving up the reusability of C++ to get the small increase in performance with C, especially when C++ can, where necessary, be written in a C programming style.

**PHP** is a common language for webpage design that is sometimes used as a scripting language in \*nix. PHP is designed for rapid website development, and as a result contains features that make it easy to link to databases, generate HTTP headers, and so forth. As a scripting language, it contains a relatively simple set of basic components that allow the programmer to quickly get up to speed, though it does have more sophisticated object-oriented features.

**Perl** was originally a file management language for Unix, but it has become well known for its use in CGI programming. CGI (Common Gateway Interface) is a term for programs that web servers can execute to allow web pages additional capabilities. Perl is great with regular expression pattern matching, which is a method for searching text. Perl can be used for databases and other useful server functions, and it is simple to pick up the basics if you have experience in any imperative language. Web hosting services prefer Perl over C++ as a CGI language because the web hosts can inspect Perl script files, since they're just text files, while C++ is compiled, so it can't be inspected for potentially dangerous code. Perl is, however, notorious for its "write once" style of code -- it's very easy to write Perl scripts taking advantage of lots of shortcuts that you later cannot understand.