Most popular programming languages:

1. Java

Java uses a compiler, and is an object-oriented language released in 1995 by Sun Microsystems. Java is the number one programming language today for many reasons. First, it is a well-organized language with a strong library of reusable software components. Second, programs written in Java can run on many different computer architectures and operating systems because of the use of the JVM ( Java virtual machine ). Sometimes this is referred to as code portability or even WORA ( write once, run anywhere ). Third, Java is the language most likely to be taught in university computer science classes. A lot of computer science theory books written in the past decade use Java in the code examples. So learning Java syntax is a good idea even if you never actually code in it.

2. C

C is a compiled, procedural language developed in 1972 by Dennis Ritchie for use in the UNIX operating system. Although designed to be portable in nature, C programs must be specifically compiled for computers with different architectures and operating systems. This helps make them lightning fast. Although C is a relatively old language, it is still widely used for system programming, writing other programming languages, and in embedded systems.

3. C++

C++ is a compiled, multi-paradigm language written as an update to C in 1979 by Bjarne Stroustrup. It attempts to be backwards-compatible with C and brings object-orientation, which helps in larger projects. Despite it's age, C++ is used to create a wide array of applications from games to office suites.

4. PHP

PHP uses a run-time interpreter, and is a multi-paradigm language originally developed in 1996 by Rasmus Lerdorf to create dynamic web pages. At first it was not even a real programming language, but over time it eventually grew into a fully featured object-oriented programming language. Although PHP has been much criticized in the past for being a bit sloppy and insecure, it's been pretty good since version 5 came out in 2004. It's hard to argue with success. Today, PHP is the most popular.

6. C#

C# is a compiled, object-oriented language written by Microsoft. It is an open specification, but rarely seen on any non-Windows platform. C# was conceived as Microsoft's premium language in its .NET Framework. It is very similar to Java in both syntax and nature.

Similarities and difference with C#:

Similarity and difference with C#/Java :

In Java methods are virtual by default but you can make them final. (In C# they're sealed by default, but you can make them virtual.)Generics are completely different between the two; Java generics are just a compile-time "trick" (but a useful one at that). In C# and .NET generics are maintained at execution time too, and work for value types as well as reference types, keeping the appropriate efficiency (e.g. a List<byte> as a byte[] backing it, rather than an array of boxed bytes.)C# doesn't have checked exceptions,Java doesn't allow the creation of user-defined value types.Java doesn't have operator and conversion overloading

Similarity and difference with C#/C :

C# object syntax is based on a fairly obvious basis of C structs.C# has operator overloading. In Objective C, all messages are explicit.C# has an enormous class library, which is rather sloppy. Objective C has a small but tightly designed and powerful class library.

Similarity and difference with C#/C++ :

Performance (speed and size): Applications built in C# may not perform as well when compared with native C++. C# has an intrusive garbage collector, reference tracking and other overheads with some of the framework services.Limitation: With C#, features like multiple inheritance from classes (C# implements a different approach called Multiple Implementation, where a class can implement more than one interface), declaring objects on the stack, deterministic destruction (allowing RAII) and allowing default arguments as function parameters (in C# versions < 4.0) will not be available.Runtime Redistribution: Programs need to be distributed with the .NET framework (pre-Windows XP or non-Windows machines), similar to the issue with the Java language, with all the normal upgrade requirements attached.

Similarity and difference with C#/PHP :

PHP can run on any platform, C# can natively ONLY RUN ON WINDOWS driven platforms.C# without aspx won't work at all, you need aspx to serve the webpages.I would use PHP if I want a minimal Linux installation and yet do not develop too complex websites. Or if I need to be hosted on some host providers which doesn't offer any alternative for ASPx.I would use C# for all web development which involve more than some simple dynamic web page. Visual Studio makes me gain a lot of time and the framework is really powerful.Debugging a PHP application is a nightmare compared to the easier C#.