

The important features of Java 1.8. C++ vs Java

- 1. Simple
- 2. Object-Oriented
- 3. Portable
- 4. Platform independent
- 5. Secured
- 6. Robust
- 7. Architecture neutral
- 8. Interpreted
- 9. High Performance
- 10. Multithreaded
- 11. Distributed
- 12. Dynamic

C++	Java
C++ is platform-dependent.	Java is platform-independent.
C++ is mainly used for system programming.	Java is mainly used for application programming. It is widely used in Windows-based, web-based, enterprise, and mobile applications.
C++ was designed for systems and applications programming. It was an extension of the C programming language	Java was designed and created as an interpreter for printing systems but later extended as a support network computing. It was designed to be easy to use and accessible to a broader audience.
C++ supports the goto statement.	Java doesn't support the goto statement.
C++ supports multiple inheritance.	Java doesn't support multiple inheritance through class. It can be achieved by using interfaces in java
C++ supports operator overloading	Java doesn't support operator overloading.
C++ supports pointers. You can write a pointer program in C++.	Java supports pointer internally. However, you can't write the pointer program in java. It means java has restricted pointer support in java.
C++ uses compiler only. C++ is compiled and run using the compiler which converts source code into machine code so, C++ is platform dependent.	Java uses both compiler and interpreter. Java source code is converted into bytecode at compilation time. The interpreter executes this bytecode at runtime and produces output. Java is interpreted that is why it is platform-independent.
C++ supports both call by value and call by reference.	Java supports call by value only. There is no call by reference in java.
C++ supports structures and unions.	Java doesn't support structures and unions.
C++ doesn't have built-in support for threads. It relies on third-party libraries for thread support.	Java has built-in thread support.
C++ doesn't support documentation comments.	Java supports documentation comment (/** ... */) to create documentation for java source code.
C++ supports virtual keyword so that we can decide whether or not to override a function.	Java has no virtual keyword. We can override all non-static methods by default. In other words, non-static methods are virtual by default.
C++ doesn't support >>> operator.	Java supports unsigned right shift >>> operator that fills zero at the top for the negative numbers. For positive numbers, it works same like >> operator.
C++ always creates a new inheritance tree.	Java always uses a single inheritance tree because all classes are the child of the Object class in Java. The Object class is the root of the inheritance tree in java.
C++ is nearer to hardware.	Java is not so interactive with hardware.
C++ is an object-oriented language. However, in the C language, a single root hierarchy is not possible.	Java is also an object-oriented language. However, everything (except fundamental types) is an object in Java. It is a single root hierarchy as everything gets derived from java.lang.Object.