

Website

<https://www.curiousinfotecholutions.com>

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Java Programming

Day-3



Difference Between

C++ Vs Java

<https://youtu.be/i9EL4TvO93s>

Email id:
info@curiousinfotecholutions.com

Syllabus Day-3

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Explore C++ vs Java.

- ✓ Platform-independent
- ✓ Mainly used for
- ✓ Design Goal
- ✓ Inheritance
- ✓ Overloading
- ✓ Pointers
- ✓ Translators
- ✓ Call by Value
- ✓ Call by Reference
- ✓ Structure
- ✓ Union
- ✓ Thread Support
- ✓ Virtual
- ✓ Near to Hardware



C++ vs Java

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Comparison Index	C++	Java
Platform-independent	C++ is platform-dependent.	Java is platform-independent.
Mainly used for	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.
Design Goal	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.
Goto	C++ supports the goto statement.	Java doesn't support the goto statement.

Contd.. C++ vs Java

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Multiple inheritance	C++ supports multiple inheritance.	Java doesn't support multiple inheritance through class. It can be achieved by using interfaces in java.
Operator Overloading	C++ supports operator overloading.	Java doesn't support operator overloading.
Pointers	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.

Contd..C++ vs Java

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Compiler and Interpreter	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.
Call by Value and Call by reference	C++ supports both call by value and call by reference.	Java supports call by value only. There is no call by reference in java.
Structure and Union	C++ supports both structures and unions.	Java doesn't support structures and unions.

Contd.. C++ vs Java

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Thread Support	C++ doesn't have built-in support for threads. It relies on third-party libraries for thread support.	Java has built-in thread support .
Documentation comment	C++ doesn't support documentation comments .	Java supports documentation comment (<code>/** ... */</code>) to create documentation for java source code.
Virtual Keyword	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 .
unsigned right shift >>>	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 .

Contd.. C++ vs Java

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Inheritance Tree	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.
Hardware	C++ is nearer to hardware.	Java is not so interactive with hardware.
Object-oriented	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 is an object in Java. It is a single root hierarchy as everything gets derived from java.lang.Object.

Java doesn't support default arguments like C++.

Java does not support header files like C++.

Java uses the import keyword to include different classes and methods



C++ Program Example

```
#include <iostream>
using namespace std;
int main() {
    cout << "Hello C++ Programming";
    return 0;
}
```

File: Simple.cpp

Java Program Example

```
class Simple{
    public static void main(String args[]){
        System.out.println("Hello Java");
    }
}
```

File: Simple.java