

# Unit – 1 Introduction to JAVA



**1.1 History of JAVA**

**1.2 Comparison of JAVA and C++**

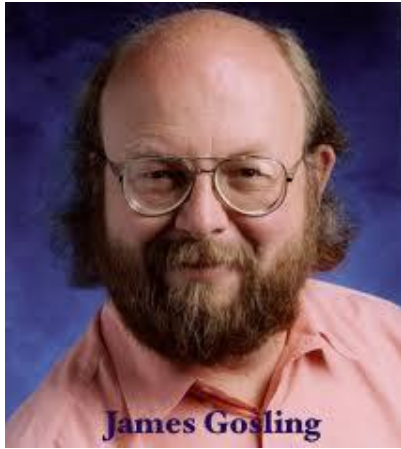
**1.3 Features**

**1.4 JAVA and Internet**

**1.5 JDK Environment**

**Prof. A. P. Chaudhari (M.Sc., SET)  
HOD, Department of Computer Science  
SVS's Dadasaheb Rawal College, Dondaicha**

## 1.1 History of JAVA



- Java was developed by the group of Sun Microsystems Engineers, lead by **James Gosling**.
- James Gosling and his team members (also known as **Green Team**), started developing a new language which was completely System Independent.
- This language was initially called OAK. They decided to rename OAK as JAVA in 1994. **JAVA was publically released first publication JAVA 1.0 on 27 May 1995.**
- In 2006-07 Sun released JAVA as open source and platform independent software.
- JAVA was designed for the development of software for consumer electronic devices like TV, VCR, Toaster and such other electronics machines.

## 1.2 Comparison of JAVA and C++

<b>C++</b>	<b>JAVA</b>
C++ is Object Oriented Programming Language.	JAVA is Pure Object Oriented Programming Language.
C++ is mainly used for system programming.	Java is mainly used for application programming.
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 interfaces in java.
C++ supports operator overloading.	Java doesn't support operator overloading.

## 1.2 Comparison of JAVA and C++

<b>C++</b>	<b>JAVA</b>
You can write pointer program in C++.	you can't write the pointer program in java.
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++ supports virtual keyword	Java has no virtual keyword.
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 compiler and interpreter both. Java source code is converted into byte code at compilation time. The interpreter executes this byte code at runtime and produces output. So, JAVA is platform independent.

## 1.3 Features

### **Simple :**

JAVA is designed to be easy for the professional programmer to learn and use effectively. It is a simple language because of its various features- JAVA inherits the C/C++ syntax and many of the object oriented features of C++. Many features of C/C++ that are redundant are not the part of JAVA.

### **Object Oriented :**

JAVA is a pure object oriented programming language due to following reasons-

- Everything in JAVA is an object.
- JAVA implements all of the features of OOPs including inheritance, polymorphism, etc.
- Most of the methods (functions) and constructors of the JAVA classes are overloaded.

## 1.3 Features

- JAVA does not support the multiple inheritance to avoid the duplication of data, but it is supported by interface.
- All program code and data reside within object and classes.

### **Distributed:**

JAVA facilitates the building of distributed application by a collection of classes for use in networked application. By using JAVA's URL class, an application can easily access a remote server. JAVA application can open and access remote objects on internet. This enables multiple programmers to use multiple locations to work together on a single project.

### **Robust:**

Robust means strong and powerful. JAVA puts a lot of important on early checking of possible errors. It makes an effort to eliminate error situations at compile time error checking and runtime checking.

## 1.3 Features

Features of JAVA like memory management, automatic garbage collection, type checking mechanism, runtime exception handling, etc. makes JAVA as robust.

### **Secure:**

Security becomes important issue for a language that is used for programming on internet. JAVA system not only verifies all memory access but also ensures that no viruses are communicated with an application or applet. It also contains security manager class that determines what resources a class can access such as reading and writing to the local disk.

### **Architecture Neutral:**

This feature makes JAVA language very special. JAVA programs can run on any platform that is they can run on different CPU's and on different operating system architecture. JAVA compiler produce byte code that can be implemented on any machine.

## 1.3 Features

### Portable:

JAVA code is portable, as new architecture are developed, the JAVA environment could be ported to them. The feature “Write once, Run anywhere” make JAVA portable. JAVA is portable because the compiler itself is written in JAVA.

### Interpreted:

JAVA language has compiler and interpreter both. This has been designed to ensure platform independence nature for the language. Due to this JAVA has been made two stage system. First, JAVA compiler translates the source code into byte code instructions for the machine does not exist. This machine is called **JAVA Virtual Machine (JVM)**. The byte code is also called as Virtual Machine code. It is not a machine specific. The machine specific code known as machine code generated by JAVA interpreter. In the second stage, JAVA interpreter generates machine code that can be directly executed by machine that is running JAVA Program.



## 1.3 Features

### **High Performance:**

The new JVM uses technology known as Just-In-Time compilation. It compiles JAVA byte code into native machine code which increase the performance. Sun recently developed the JAVA HotSpot performance engine, which includes a compiler for optimizing the frequently used code. The HotSpot performance engine can be plugged into the JVM to increase JAVA performance.

**Multithreading:** JAVA supports multithreaded programming which allows us to write a program that can perform more than one task simultaneously. User need not wait for one program to finish one task before starting next task.

**Dynamic:** It is dynamic in nature that is JAVA program can link to new class libraries, objects, methods etc. at runtime. JAVA language also provide the facility to include the functions of other languages like C and C++. They are referred as native methods.

### 1.4 JAVA and Internet

JAVA is strongly associated with the internet. Internet users can use JAVA to create applet programs and run them locally using a “Java Enabled Browser” such as HotJava. They can also use a Java Enabled Browser to download an applet located on a computer anywhere in the internet and run it on his local computer. In fact, Java applets have made the internet a true extension of the storage system of the local computer. Internet users can also setup their websites containing java applets that could be used by other remote users of internet.

## 1.5 JDK Environment

JAVA environment includes a number of development tools, classes and methods. The development tools are the part of the system known as Java Development Kit (JDK). The classes and methods are part of the Java Standard Library (JSL) also known as Application Programming Interface (API).

The JDK comes with set of tools that are used for developing running Java program. JDK has following tools for Java development –

### **i) Javac:**

The Java compiler that translate human readable java source code to architecture neutral byte code which java interpreter can understand.

### **ii) Java:**

Java interpreter that runs applets and applications interpreting byte code files. It launches a java application by starting java runtime environment, loading a specified class and invoking that class main method to run the program.

## 1.5 JDK Environment

### iii) Applet Viewer:

This enables us to run java applets. Normally, applets are the programs those can be embedded in a web page and run in web browser. This is very useful tool which gives the same effect but on console.

### iv) JavaDoc:

It creates HTML format documentation from java source code file.