**1.Introduction to Java Programming**

**1.1 Overview of Java :-**

History of Java

James Gosling initiated Java language project in June 1991 for use in one of his many set-top box projects. The language, initially called ‘Oak’ after an oak tree that stood outside Gosling's office, also went by the name ‘Green’ and ended up later being renamed as Java, from a list of random words.

Sun released the first public implementation as Java 1.0 in 1995. It promised Write Once, Run Anywhere (WORA), providing no-cost run-times on popular platforms.

On 13 November, 2006, Sun released much of Java as free and open source software under the terms of the GNU General Public License (GPL).

On 8 May, 2007, Sun finished the process, making all of Java's core code free and open-source, aside from a small portion of code to which Sun did not hold the copyright.

Tools You Will Need

For performing the examples discussed in this tutorial, you will need a Pentium 200-MHz computer with a minimum of 64 MB of RAM (128 MB of RAM recommended).

You will also need the following softwares −

* Linux 7.1 or Windows xp/7/8 operating system
* Java JDK 8
* Microsoft Notepad or any other text editor

This tutorial will provide the necessary skills to create GUI, networking, and web applications using Java.

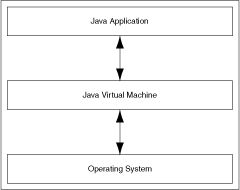
What is Next?

The next chapter will guide you to how you can obtain Java and its documentation. Finally, it instructs you on how to install Java and prepare an environment to develop Java applications.

Java programming language was originally developed by Sun Microsystems which was initiated by James Gosling and released in 1995 as core component of Sun Microsystems' Java platform (Java 1.0 [J2SE]).

The latest release of the Java Standard Edition is Java SE 8. With the advancement of Java and its widespread popularity, multiple configurations were built to suit various types of platforms. For example: J2EE for Enterprise Applications, J2ME for Mobile Applications.

The new J2 versions were renamed as Java SE, Java EE, and Java ME respectively. Java is guaranteed to be Write Once, Run Anywhere.



## Overview of Java

Java has emerged as the object-oriented programming language of choice. Some of the important concepts of Java include:

* A Java virtual machine (JVM), which provides the fundamental basis for platform independence
* Automated storage management techniques, such as garbage collection
* Language syntax that is similar to that of the C language

The result is a language that is object-oriented and efficient for application programming.

This section covers the following topics:

* [Java and Object-Oriented Programming Terminology](https://docs.oracle.com/en/database/oracle/oracle-database/12.2/jjdev/Java-overview.html#GUID-2179A214-AF5E-4FE5-83DF-EBB6A464ED35)
* [Key Features of the Java Language](https://docs.oracle.com/en/database/oracle/oracle-database/12.2/jjdev/Java-overview.html#GUID-7E0D6882-64EA-43C2-B5ED-E89CD62B8178)
* [Java Virtual Machine](https://docs.oracle.com/en/database/oracle/oracle-database/12.2/jjdev/Java-overview.html#GUID-5FE3D958-3D58-4976-B5E2-1B13F7B1BF47)
* [Java Class Hierarchy](https://docs.oracle.com/en/database/oracle/oracle-database/12.2/jjdev/Java-overview.html#GUID-D20F4AEA-5524-41F2-AF2B-B17590B70FF0)

### 1.1.1 Java and Object-Oriented Programming Terminology

The following terms are common in Java application development in Oracle Database environment:

* [Classes](https://docs.oracle.com/en/database/oracle/oracle-database/12.2/jjdev/Java-overview.html#GUID-C1756D87-6CAD-4C87-A498-E7C996077D9A)
* [Objects](https://docs.oracle.com/en/database/oracle/oracle-database/12.2/jjdev/Java-overview.html#GUID-061CB7CD-144F-4B3C-9409-748B94C25A09)
* [Interfaces](https://docs.oracle.com/en/database/oracle/oracle-database/12.2/jjdev/Java-overview.html#GUID-DB2DCB58-35CD-43F8-B6EF-544B4A4ED23A)
* [Encapsulation](https://docs.oracle.com/en/database/oracle/oracle-database/12.2/jjdev/Java-overview.html#GUID-68EE1A7B-1F78-4074-AB76-AF9B2CE878F6)
* [Inheritance](https://docs.oracle.com/en/database/oracle/oracle-database/12.2/jjdev/Java-overview.html#GUID-5C62367E-3197-4E67-A38E-39CE04C7B795)
* [Polymorphism](https://docs.oracle.com/en/database/oracle/oracle-database/12.2/jjdev/Java-overview.html#GUID-9AC04464-EE79-4890-87DA-507523A8DFBA)

### Application of Java

Java is widely used in every corner of world and of human life. Java is not only used in softwares but is also widely used in designing hardware controlling software components. There are more than 930 million JRE downloads each year and 3 billion mobile phones run java.

Following are some other usage of Java :

1. Developing Desktop Applications
2. Web Applications like Linkedin.com, Snapdeal.com etc
3. Mobile Operating System like Android
4. Embedded Systems
5. Robotics and games etc.

**Assembly languages**

Software techniques started with the assembly languages, that were close to machine instruction and were easy to convert into executable code. Each hardware had its own assembly language. Assembly language contains low level instructions like move data from memory to hardware registers, do arithmetic operations, and move data back to memory. Programmers had to know the detailed architecture of the computer in order to write programs.

**Procedural languages**

After the assembly languages, high level languages were developed. Here the language compiler is used to convert the high level program to machine instructions, freeing the programmers from the burden of knowing the computer hardware architecture. To promote the re-use of code and to minimize the use of GOTO instructions, "procedural" techniques were introduced. This simplified the creation and maintenance of software control flow, but left out the organization of data. It became a nightmare to debug and maintain programs having many global variables (i.e. variables that contain data that can be modified anywhere in the application).