

Study on Java Language

Sayalee W. Sangle

Computer Science Department, Sinhgad Institute Of Business Admin. &
Research Kondhwa, Pune, Savitribai Phule Pune University
Pune, India
sayaleesangle@gmail.com

Abstract :- Java is one of many high-level programming languages that use abstraction (reducing and factoring details) to create software for computers and many other electronic devices. Compared to low-level programming languages, high-level programming languages use a mixture of natural language (any language used by humans) and typical computer terms to create an easier to understand and simpler form of programming.

I. INTRODUCTION

Java is a general-purpose computer programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to byte code that can run on any Java virtual machine (JVM) regardless of computer architecture. As of 2016, Java is one of the most popular programming language in use particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by James Gosling at Sun Microsystems (which has since been acquired by Oracle Corporation) and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++, but it has fewer low level facilities than either of them.

II. HISTORY

Java was started as a project called "Oak" by James Gosling in June 1991. Gosling's goals were to implement a virtual machine and a language that had a familiar C like notation but with greater uniformity and simplicity than C/C++. The First publication of Java 1.0 was released by Sun Microsystems in 1995. I made the promise of "Write Once Run Anywhere", with free runtimes on popular platforms. In 2006-2007 Sun released java as a open source and platform independent software. Over time new enhanced versions of java have been released. Java releases:

1. Initial Java Versions 1.0 and 1.1 was released in the year 1996 for Linux, Solaris, Mac and Windows.
2. Java Versions 1.2 (Commonly called as java 2) was released in the year 1998.

3. Java Versions 1.3 codename 'Kestrel' was released in the year 2000.
4. Java Versions 1.4 codename 'Merlin' was released in the year 2002.
5. Java Versions 1.5/Java SE 5 codename 'Tiger' was released in the year 2004.
6. Java Versions 1.6/Java SE 6 codename 'Mustang' was released in the year 2006.
7. Java Versions 1.7/Java SE 7 codename 'Dolphin' was released in the year 2011.
8. Java Versions 1.8 is the current stable release was released in 2015.

FEATURES OF JAVA

1) Simple: Java is a simple language because of its various features, Java Doesn't Support Pointers, Operator Overloading etc. It doesn't require unreferenced object because java support automatic garbage collection. Java provides bug free system due to the strong memory management.

2) Object-Oriented:

Object-Oriented Programming Language (OOPs) is the methodology which provide software development and maintenance by using object state, behaviour, and properties. Object Oriented Programming Language must have the following characteristics.

1) Encapsulation 2) Polymorphism 3) Inheritance 4) Abstraction

As the languages like Objective C, C++ fulfills the above four characteristics yet they are not fully object oriented languages because they are structured as well as object oriented languages.

In java everything is an Object. Java can be easily extended since it is based on the Object model

3) Secure

Java is Secure Language because of its many features it enables to develop virus-free, tamper-free systems. Authentication techniques are based on public-key encryption. Java does not support pointer explicitly for the memory.

All Program Run under the sandbox.

4) Robust

Java was created as a strongly typed language. Data type issues and problems are resolved at compile-time, and implicit casts of a variable from one type to another are not allowed. Memory management has been simplified java in two ways. First Java does not support direct pointer manipulation or arithmetic. This make it possible for a java program to overwrite memory or corrupt data.

Second, Java uses runtime garbage collection instead of instead of freeing of memory. In languages like c++, it Is necessary to delete or free memory once the program has finished with it.

5) Platform-independent.

Java Language is platform-independent due to its hardware and software environment. Java code can be run on multiple platforms e.g. windows, Linux, sun Solaris, Mac/Os etc. Java code is compiled by the compiler and converted into byte code. This byte code is a platform independent code because it can be run on multiple platforms i.e. Write Once and Run Anywhere (WORA).

6) Architecture neutral

It is not easy to write an application that can be used on Windows , UNIX and a Macintosh. And its getting more complicated with the move of windows to non Intel CPU architectures.

Java takes a different approach. Because the Java compiler creates byte code instructions that are subsequently interpreted by the java interpreter, architecture neutrality is achieved in the implementation of the java interpreter for each new architecture.

7) Portable

Java code is portable. It was an important design goal of Java that it be portable so that as new architectures (due to hardware, operating system, or both) are developed, the java environment could be ported to them. In java, all primitive types (integers, longs, floats, doubles, and so on) are of defined sizes, regardless of the machine or operating system on which the program is run. This is in direct contrast to languages like C

and C++ that leave the sized of primitive types up to the compiler and developer.

Additionally, Java is portable because the compiler itself is written in Java.

8) Dynamic

Because it is interpreted , Java is an extremely dynamic language, At runtime, the java environment can extends itself by linking in classes that may be located on remote servers on a network(for example, the internet) At runtime, the java interpreter performs name resolution while linking in the necessary classes. The Java interpreter is also responsible for determining the placement of object in memory. These two features of the Java interpreter solve the problem of changing the definition of a class used by other classes.

9) Interpreted

We all know that Java is an interpreted language as well. With an interpreted language such as Java, programs run directly from the source code. The interpreter program reads the source code and translates it on the fly into computations. Thus, Java as an interpreted language depends on an interpreter program. The versatility of being platform independent makes Java to outshine from other languages. The source code to be written and distributed is platform independent Another advantage of Java as an interpreted language is its error debugging quality. Due to this any error occurring in the program gets traced. This is how it is different to work with Java.

10) High performance

For all but the simplest or most infrequently used applications, performance is always a consideration for most applications, including graphics-intensive ones such as are commonly found on the world wide web, the performance of java is more than adequate.

11) Multithreaded

Writing a computer program that only does a single thing at a time is an artificial constraint that we've lived with in most programming languages. With java, we no longer have to live with this limitation. Support for multiple, synchronized threads is built directly into the Java language and runtime environment. Synchronized threads are extremely useful in creating distributed, network-aware applications. Such as application may be communicating with a remote server in one thread while interacting with a user in a different thread.

12) Distributed.

Java facilitates the building of distributed application by a collection of classes for use in networked applications. By

using java's URL (Uniform Resource Locator) class, an application can easily access a remote server. Classes also are provided for establishing socket-level connections.

USES OF JAVA

Java technology is a high-level programming and a platform independent language. Java is designed to work in the distributed environment on the Internet. Java has a GUI features that provides you better "look and feel" over the C++ language, moreover it is easier to use than C++ and works on the concept of object-oriented programming model. Java enable us to play online games, video, audio, chat with people around the world, Banking Application, view 3D image and Shopping Cart. Java find its extensive use in the intranet applications and other e-business solutions that are the grassroots of corporate computing. Java, regarded as the most well described and planned language to develop an applications for the Web. Java is a well known technology which allows you for software designed and written only once for an "virtual machine" to run on a different computers, supports various Operating System like Windows PCs, Macintoshes, and Unix computers. On the web aspect, Java is popular on web servers, used by many of the largest interactive websites. Java is used to create standalone applications which may run on a single computer or in distributed network. It is also be used to create a small application program based on applet, which is further used for Web page. Applets make easy and possible to interact with the Web page.

Types of Applications that Run on Java

1. Desktop GUI Applications:

Java provides GUI development through various means like Abstract Windowing Toolkit (AWT), Swing and JavaFX. While AWT contains a number of pre-constructed components such as menu, button, list, and numerous third-party components, Swing, a GUI widget toolkit, additionally provides certain advanced components like trees, tables, scroll panes, tabbed panel and lists. JavaFX, a set of graphics and media packages, provides Swing interoperability, 3D graphic features and self-contained deployment model which facilitates quick scripting of Java applets and applications.

2. Mobile Applications:

Java Platform, Micro Edition (Java ME or J2ME) is a cross-platform framework to build applications that run across all Java supported devices, including feature phones and smart phones. Further, applications for Android, one of the most popular mobile operating systems, are usually scripted in Java using the Android Software Development Kit (SDK) or other environments.

3. Embedded Systems:

Embedded systems, ranging from tiny chips to specialized computers, are components of larger electromechanical systems performing dedicated tasks. Several devices, such as SIM cards, blue-ray disk players, utility meters and televisions, use embedded Java technologies. According to Oracle, 100% of Blu-ray Disc Players and 125 million TV devices employ Java.

4. Web Applications:

Java provides support for web applications through Servlets, Struts or JSPs. The easy programming and higher security offered by the programming language has allowed a large number of government applications for health, social security, education and insurance to be based on Java. Java also finds application in development of eCommerce web applications using open-source eCommerce platforms, such as Broadleaf.

5. Web Servers and Application Servers:

The Java ecosystem today contains multiple Java web servers and application servers. While Apache Tomcat, Simple, Jo!, Rimfaxe Web Server (RWS) and Project Jigsaw dominate the web server space, WebLogic, WebSphere, and Jboss EAP dominate commercial application server space.

6. Enterprise Applications:

Java Enterprise Edition (Java EE) is a popular platform that provides API and runtime environment for scripting and running enterprise software, including network applications and web-services. Oracle claims Java is running in 97% of enterprise computers. The higher performance guarantee and faster computing in Java has resulted in high frequency trading systems like Murex to be scripted in the language. It is also the backbone for a variety of banking applications which have Java running from front user end to back server end.

7. Scientific Applications:

Java is the choice of many software developers for writing applications involving scientific calculations and mathematical operations. These programs are generally considered to be fast and secure, have a higher degree of portability and low maintenance. Applications like MATLAB use Java both for interacting user interface and as part of the core system. In conclusion, Java is widely applicable across different types of applications. It offers cross-functionality and portability, and these features, among many others, make Java the programming language of choice for software development of a specific nature.

REFERENCES

- [1] Wikipedia Java Programming
- [2] Gosling, James, Joy, Bill, Steele, Guy L., Jr., Bracha, Gilad(2005).The Java Language Specification(3rd ed.)Addison-Wesley. ISBN 0-321-24678-0.
- [3] Java Grande Forum, <http://www.javagrande.org>
- [4] M.Bornemann,R. V. V.Nieuwpoort, T. Kielmann,MPJ/Ibis: a Flexible and Efficient Message Passing Platform for Java, in:Poc. 12th European PVM/MPI Users' Group Meeting(EuroPVM/MPI'05),LNCS vol.3666,Sorrento,Italy, 2005,pp.217-224
- [5] Vipul Sawhney. 2003: Developing Next Generation LBS applications using the LocationNet Platform