

JAVA: THINGS TO KNOW ABOUT

Java is a programming language and computing platform first released by Sun Microsystems in 1995. There are lots of applications and websites that will not work unless you have Java installed, and more are created every day. Java is fast, secure, and reliable. From laptops to datacenters, game consoles to scientific supercomputers, cell phones to the Internet, Java is everywhere!



Java platform is a collection of programs that help to develop and run programs written in the Java programming language. Java platform includes an execution engine, a compiler, and a set of libraries. Java is platform-independent language. It is not specific to any processor or operating system. Java produces applets (browser-run programs), which facilitate graphical user interface (GUI) and object interaction by Internet users. Prior to Java applets, Web pages were typically static and non-interactive.

Java applets have diminished in popularity with the release of competing products, such as Adobe Flash and Microsoft Silverlight. Java applets run in a Web browser with Java Virtual Machine (JVM), which translates Java bytecode into native processor instructions and allows indirect OS or platform program execution. JVM provides the majority of components needed to run bytecode, which is usually smaller than executable programs written through other programming languages. Bytecode cannot run if a system lacks required JVM.





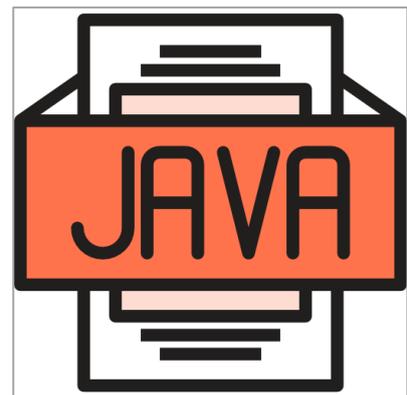
Java program development requires a Java software development kit (SDK) that typically includes a compiler, interpreter, documentation generator and other tools used to produce a complete application. Development time may be accelerated through the use of integrated development environments (IDE) - such as JBuilder, Netbeans, Eclipse or JCreator. IDEs facilitate the development of GUIs, which include buttons, text boxes, panels, frames, scrollbars and other objects via drag-and-drop and point-and-click actions.

Take away

Java platform is a collection of programs that help to develop and run programs written in the Java programming language. Java programs are found in desktops, servers, mobile devices, smart cards and Blu-ray Discs

History of Java

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. Oracle Corporation is the current owner of the official implementation of the Java SE platform, following their acquisition of Sun Microsystems on January 27, 2010. This implementation is based on the original implementation of Java by Sun. The Oracle implementation is available for Microsoft Windows, Mac OS X, Linux and Solaris.



The Oracle implementation is packaged into two different distributions:

1. **Java Runtime Environment (JRE)** which contains the parts of the Java SE platform required to run Java programs and is intended for end users.



2. **Java Development Kit (JDK)** which is intended for software developers and includes development tools such as the Java compiler, Java.doc, Jar, and a debugger.



Java platforms

There are three key platforms upon which programmers develop Java applications:

1. **Java SE-** Simple, stand-alone applications are developed using Java Standard Edition. Formerly known as J2SE, Java SE provides all of the APIs needed to develop traditional desktop applications.
2. **Java EE-** The Java Enterprise Edition, formerly known as J2EE, provides the ability to create server-side components that can respond to a web-based request-response cycle. This arrangement allows the creation of Java programs that can interact with Internet-based clients, including web browsers, CORBA-based clients and even REST- and SOAP-based web services.
3. **Java ME-** Java also provides a lightweight platform for mobile development known as Java Micro Edition, formerly known as J2ME.



Java ME has proved a prevalent platform for embedded device development, but it struggled to gain traction in the smartphone development arena.

Take away

Java software has come a long way. More than half of all handheld phones in the world run on Android, giving Java an incredibly strong hold in the smartphone market.

Main uses of Java

It is easy for developers to write programs which employ popular software design patterns and best practices using the various components found in Java EE. For example, frameworks such as Struts and Java Server Faces all use a Java servlet to implement the front controller design pattern for centralizing requests.

A big part of the Java ecosystem is the large variety of open source and



community-built projects, software platforms and APIs. For example, the Apache Foundation hosts a variety of projects written using Java, including simple logging frameworks for Java (SLF4J), both Yarn and Hadoop processing

frameworks, Microservices development platforms and integration platforms.

In terms of mobile development, Java is commonly used as the programming language for Android applications. Java tends to be preferred by Android developers because of Java's security, object-oriented paradigms, regularly updated and maintained feature sets, use of JVM and frameworks for networking, IO and threading.

Although Java is widely used, it still has fair criticisms. Java syntax is often criticized for being too verbose. In response, several peripheral languages have emerged to address these issues, including Groovy. Due to the way Java references objects internally, complex and concurrent list-based operations slow the JVM. The Scala language addresses many of the shortcomings of the Java language that reduce its ability to scale.



TO SUM UP: WHY CHOOSE JAVA

1. **Ease of Use:** The fundamentals of Java came from a programming language called C++. Although C++ is a powerful language, it is complex in its syntax and inadequate for some of Java's requirements.
2. **Reliability:** Java needed to reduce the likelihood of fatal errors from programmer mistakes. With this in mind, object-oriented programming was introduced.
3. **Security:** Because Java was originally targeting mobile devices that would be exchanging data over networks, it was built to include a high level of security.
4. **Platform Independence:** Programs need to work regardless of the machines they're being executed on. Java was written to be a portable and cross-platform language that doesn't care about the operating system, hardware, or devices that it's running on.

Java's popularity can be traced to it being a robust, secure, easy to use, and portable programming language.