# Introduction to Java

## I. INTRODUCTION to Java

Java is the general name of Java object-oriented programming language and Java platform introduced by Sun Microsystems company in May 1995. It was jointly developed by James Gosling and colleagues and officially launched in 1995.

## II. Oracle

After Sun Microsystems was acquired by Oracle, Java also became the product of Oracle.

The relational database product developed by Oracle is famous for its excellent performance. Many large websites also use Oracle system, which is the best database product in the world.



## III. Java Platform System

JavaSE(J2SE)(Platform Standard Edition, Platform Standard Edition) including Java basics, Java object-oriented, and advanced Java language.

JavaEE(J2EE)(Platform Enterprise Edition, Platform Enterprise Edition) includes: javaweb, Java framework, microservices, and distributed, middleware.

JavaME(J2ME)(Platform Micro Edition, Platform Mini Edition) JavaEE is understandable, so there is no need to go into it.

## IV. Main features of Java language



#### 1. The Java language is simple

The syntax of Java language is very close to C language and C ++ language, making it easy for most programmers to learn and use. On the other hand, Java discards the features that are rarely used, difficult to understand, and confusing in C ++, such as operator overloading, multi-inheritance, and automatic forced type conversion. In particular, the Java language does not use pointers, but references. It also provides automatic allocation and recycling of memory space so that programmers do not have to worry about memory management.

#### 2. The Java language is object-oriented

The Java language provides object-oriented features such as classes, interfaces, and inheritance. For simplicity, it only supports single inheritance between classes, but supports multiple inheritance between interfaces, and supports the implementation mechanism between classes and interfaces (keyword: implements). Java fully supports dynamic binding, while C ++ only uses dynamic binding for virtual functions. In short, Java is a pure object-oriented programming language.

#### 3. The Java language is distributed

The Java language supports the development of Internet applications. In the basic Java application programming interface, there is a network application programming interface (java net), which provides class libraries for network application programming, including URL, URLConnection, Socket, ServerSocket, etc. The RMI (remote method activation) mechanism of Java is also an important means to develop distributed applications.

#### 4. The Java language is robust

Java's strong-type mechanism, exception handling, and automatic garbage collection are important guarantees for the robustness of Java programs. Discarding the pointer is a wise choice for Java. Java's security check mechanism makes Java more robust.

#### 5. The Java language is secure

Java is usually used in network environments. Therefore, Java provides a security mechanism to prevent malicious code attacks. In addition to many security features of the Java language, Java has a security mechanism (class ClassLoader) for classes downloaded through the network. For example, it allocates different namespaces to prevent the local class with the same name and byte code check. It also provides a security management mechanism (class SecurityManager) to enable Java applications to set security guards.

#### 6. The Java language is architecture-neutral

JThe ava program (a file with a suffix of java) is compiled into an architecture-neutral byte code format (a file with a suffix of class) on the Java platform, and can then be run in any system that implements the Java platform. This approach is suitable for heterogeneous network environments and software distribution.

#### 7. Java language is Portable

This portability comes from the neutrality of the architecture. In addition, Java strictly stipulates the length of each basic data type. The Java system itself is also very portable. The Java compiler is implemented in Java and the Java Runtime Environment is implemented in ANSI C.

#### 8. Java is Interpretive

As mentioned earlier, Java programs are compiled into byte codes on the Java platform and can then be run in any system that implements the Java platform. At runtime, the Java interpreter in the Java platform interprets and executes these bytecodes, and the classes required during execution are loaded into the runtime environment during the join phase.

#### 9. Java is high-performance

Java is indeed high-performance compared to those explanatory advanced scripting languages. In fact, the running speed of Java is closer to that of C ++ with the development of JIT(Just-In-Time) compiler technology.

#### 10. The Java language is multi-threaded

In Java, a Thread is a special object that must be created by the Thread class or its child (grandson) class. Any thread has its own run method, which contains the code to be run by the thread. Thread activity is controlled by a set of methods. The Java language supports simultaneous execution of multiple threads and provides a synchronization mechanism between multiple threads (keyword: synchronized).

#### 11. The Java language is dynamic

One of the design goals of the Java language is to adapt to a dynamically changing environment. The classes required by Java programs can be dynamically loaded into the runtime environment, or the required classes can be loaded through the network. This is also conducive to software upgrades. In addition, classes in Java have a runtime representation that allows you to check the runtime type.

## V. Development history



* On May 23rd, 1995, the Java language was born.

* In January 1996, the first JDK-JDK1.0 was born.

* In April 1996, 10 major operating system suppliers declared that they would embed JAVA technology into their products.

* In September 1996, about 83000 web pages were made using JAVA technology.

* JDK1.1 was released on February 18, 1997.

* On April 2, 1997, the JavaOne conference was held with more than 10,000 participants, setting a record for the scale of similar conferences around the world at that time.

* In September 1997, JavaDeveloperConnection more than 100,000 community members.

* In February 1998, JDK1.1 was downloaded more than 2,000 times.

* On December 8, 1998, JAVA2 enterprise platform J2EE was released

* in June 1999, SUN released three versions of Java: the standard version (JavaSE, which used to be J2SE), the Enterprise version (JavaEE used to be J2EE), and the mini version (JavaME, which used to be J2ME).

* JDK1.3 was released on May 8, 2000.

* JDK1.4 was released on May 29, 2000.

* On June 5, 2001, NOKIA announced that it would sell 0.1 billion Java-enabled mobile phones by 2003.

* J2EE1.3 was released on September 24, 2001.

* On February 26, 2002, J2SE1.4 was released. Since then, Java has greatly improved its computing power.

* In December, 2006, SUN released JRE6.0

* On April 20, 2009, Oracle purchased Sun for $7.4 billion and obtained the copyright of Java.

* In November 2010, Apache threatened to quit JCP because Oracle was unfriendly to the Java community.

* On July 28, 2011, Oracle released the official version of Java7.0.

* On March 18, 2014, Oracle released Java 8

* on September 21, 2017, Oracle released Java 9

* on March 21, 2018, Oracle released Java 10

* September 25, 2018, Java 11

* march 20, 2019, Java 12

* 2019-09-17,Java 13

* 2020-03-17,Java 14

* 2020-09,Java 15

* 2021-03，Java 16

* 2021-09，Java 17

* March 22, 2022 ,Java 18

* october 17, 2022 *,* Java 19