The Java™

Language Version History

**History**

The Java™ programming language is a general-purpose, concurrent, class-based, strongly typed, high-level, object-oriented language. It is related to C and C++ but is organized rather differently, with a number of aspects of C and C++ omitted and a few ideas from other languages included. It is intended to be a production language, not a research language.

James Gosling, Mike Sheridan, and Patrick Naughton initiated the Java language project in June 1991. Java was originally designed for interactive television, but it was too advanced for the digital cable television industry at the time. The language was initially called *Oak* after an oak tree that stood outside Gosling's office; it went by the name *Green* later, and was later renamed *Java*, from Java coffee, said to be consumed in large quantities by the language's creators. Below is the list of feature or final product releases of Java.

**Note:** Java remains a **de facto** (means "in practice but not necessarily established by law") standard, controlled through the Java Community Process (JCP).

**Table 1** – Releases of Java

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Code Name** | **Release Date** | **Remark** |
| Oak | NIL |  | Initiated at June, 1991 |
| JDK 1.0 | Oak | January 23, 1996 | **Oak** was **renamed** Java in 1994 after a trademark search revealed that Oak was used by Oak Technology. |
| JDK 1.1 | NIL | February 19, 1997 |  |
| J2SE 1.2 | Playground | December 8, 1998 | As of this version, the name "Java 2" and the version name "J2SE" (Java 2 Platform, Standard Edition) replaced JDK to distinguish the base platform from J2EE (Java 2 Platform, Enterprise Edition) and J2ME (Java 2 Platform, Micro Edition).  This was a very significant release of Java as it tripled the size of the Java platform to 1520 classes in 59 packages. |
| J2SE 1.3 | Kestrel | May 8, 2000 |  |
| J2SE 1.4 | Merlin | February 6, 2002 |  |
| J2SE 5.0 | Tiger | September 30, 2004 |  |
| Java SE 6 | Mustang | December 11, 2006 | As of this version, Sun replaced the name "J2SE" with "Java SE" and dropped the ".0" from the version number. Internal numbering for developers remains 1.6.0 |
| Java SE 7 | Dolphin | July 28, 2011 |  |
| Java SE 8 |  | Expected in Mar’14 |  |
| Java SE 9 |  | Expected in 2016 |  |
| Java SE 10 |  | Expected in 2018 |  |

**Table 2** – Language Change Details

| **Version** | **Major Language/API Changes** |
| --- | --- |
| Oak |  |
| JDK 1.0 |  |
| JDK 1.1 | * Introduced Inner Classes * JDBC API (Added) * RMI API (Added) * Reflection API (Added) * JavaBeans API (Added) |
| J2SE 1.2 | * Collections framework (Added) * Java IDL (an implementation of IDL for CORBA) * Java Plug-in for Applet (Added) * Swing API (integrated with core classes). * strictfp keyword (added) * JVM equipped with JIT compiler * HostSpot JVM shipped as an extension |
| J2SE 1.3 | * RMI (updated to support CORBA) * JNDI (included in core libraries) * JPDA (Added) * HostSpot JVM included * JavaSound API * Java Web Start added as an extension |
| J2SE 1.4 | * assert keyword (Added) * Exception chaining (re-throw/bundling original exception) * NIO (new IO or non-blocking IO) * Logging API (Added) * JAXP API (DOM and SAX) for processing XML and XSLT * Integrated JCE, JSSE and JAAS * Preferences API (Added) * Java Web Start included in core libraries * Support for IPV6 |
| J2SE 5.0 | * Generics * Enumerations * Annotations * Varargs (Ellipses) * Auto boxing/un-boxing * Static imports * Enhanced for-each loop * Concurrency Utilities * Scanner class * RMI (auto stub generation) (Updated) |
| Java SE 6 | * Java Compiler API (Added) * JDBC (Upgraded to 4.0) * JAXB; added support for StAX (Streaming API for XML) model (Upgraded to 2.0) * Annotations (support for pluggable annotations) (Updated) * Web services support through JAX-WS in Java SE (Added in Java SE; earlier it was available only in Java EE 5) |
| Java SE 7 | * Fork/Join framework (Parallel processing of a task) (Added) * Automatic Resource Management (ARM) in try statement (introduced try-with-resources statement) * Catching multiple exception types and re-throwing exceptions with improved type checking. * String constants in switch statement * Improved compiler warnings and errors when using non-reifiable formal parameters with Varargs Methods (Added @SafeVarargs annotation) * Binary literals (Add the prefix 0b or 0B to specify a binary literal) * Allowing underscores (\_) in numeric literals (e.g. 10\_000\_000) to separate groups of digits to improve readability. * Diamond operator (<>) to improve type inference for generic instance creation (e.g.: new HashMap<>()) * Merge Sort is replaced by Tim Sort to sort arrays using Arrays.sort(). * Supporting Dynamically Typed Languages on the Java Platform with the new invokedynamic JVM instruction. |
| Java SE 8 | * Lambda expressions and Method References (Anonymous inner functions) * Parallel array sorting (Arrays.parallelSort() uses the implementation of Fork/Join framework introduced in Java SE 7) * Handle Frequent HashMap Collisions with Balanced Trees (Enhancement) * Repeating annotations and Type annotations (Annotations API Enhancement) * Date-Time API (java.time) * Neshorn JavaScript engine (replacing Rhino) * Removed apt (Annotation Processing Tool) * Removed JDBC-ODBC bridge * Removed old Java Plug-in |
| Java SE 9 | * Money and Currency API (New) * Modularization of JDK; the Java platform will become modular so that applications can be installed with only the components needed) |
| Java SE 10 | * Removal of primitive types to make the Java language as fully Object Oriented. * Move from 32-bit to 64-bit addressable arrays to support large data sets. |