



Oracle Releases Java 22

New release delivers 12 JDK Enhancement Proposals to improve the Java language and enhance the development platform's performance, stability, and security

JavaOne returns to the San Francisco Bay Area in 2025

AUSTIN, Texas – March 19, 2024 – Oracle today announced the availability of [Java 22](#), the latest version of the world's number one programming language and development platform. Java 22 (Oracle JDK 22) delivers thousands of performance, stability, and security improvements to help developers increase productivity, drive innovation, and accelerate growth across their organizations. These include enhancements to the Java language, its APIs and performance, and the tools included in the Java Development Kit (JDK).

“The new enhancements in Java 22 enable more developers to quickly and easily build and deliver feature-rich, scalable, and secure applications to help organizations across the globe grow their businesses,” said Georges Saab, senior vice president, Oracle Java Platform and chair, OpenJDK governing board. “By delivering enhancements that streamline application development and extend Java’s reach to make it accessible to developers of all proficiency levels, Java 22 will help drive the creation of a wide range of new applications and services for organizations and developers alike.”

The latest JDK provides updates and improvements with 12 JDK Enhancement Proposals (JEPs). JDK 22 delivers language improvements from OpenJDK Project Amber (Statements before `super[...]`, Unnamed Variables & Patterns, String Templates, and Implicitly Declared Classes and Instance Main Methods); enhancements from Project Panama (Foreign Function & Memory API and Vector API); features related to Project Loom (Structured Concurrency and Scoped Values); core libraries and tools capabilities (Class-File API, Launch Multi-File Source-Code Programs, and Stream Gatherers); and performance updates (Region Pinning for G1).

“After nearly three decades, Java’s ability to support complex development tasks that span a wide range of use cases makes the platform as relevant as it has ever been,” said Arnal Dayaratna, research vice president, software development, IDC. “Java’s versatility and comprehensive toolset enables it to support the development of production-grade, mission-critical applications at scale, which positions it as a key enabling technology for innovative use cases such as generative AI.”

Significant updates delivered in Java 22 are:

Project Amber Features

- [JEP 447: Statements before `super\(...\)`](#): Gives developers the freedom to express the behavior of constructors. By allowing statements that do not reference the instance being created to appear before an explicit constructor invocation, this feature enables a more natural placement of logic that needs to be factored into auxiliary static methods, auxiliary

intermediate constructors, or constructor arguments. It also preserves the existing assurance that constructors run in top-down order during class instantiation, helping ensure that code in a subclass constructor cannot interfere with superclass instantiation. In addition, this feature does not require any changes to the Java Virtual Machine (JVM) and relies only on the current ability of the JVM to verify and execute code that appears before explicit constructor invocations within constructors.

- [JEP 456: Unnamed Variables & Patterns](#): Helps improve developer productivity by enhancing the Java language with unnamed variables and patterns, which can be used when variable declarations or nested patterns are required but never used. This reduces opportunities for error, improves the readability of record patterns, and increases the maintainability of all code.
- [JEP 459: String Templates \(Second Preview\)](#): Simplifies the development of Java programs by making it easy to express strings that include values computed at run time, while also improving the security of programs that compose strings from user-provided values and pass them to other systems. Additionally, the readability of expressions mixed with text is enhanced, and non-string values computed from literal text and embedded expressions can be created without having to transit through an intermediate string representation.
- [JEP 463: Implicitly Declared Classes and Instance Main Methods \(Second Preview\)](#): Helps accelerate learning by offering a smooth on-ramp to Java programming to enable students to write their first programs without needing to understand language features designed for large programs. With this feature, educators can introduce concepts in a gradual manner and students can write streamlined declarations for single-class programs and seamlessly expand their programs to use more advanced features as their skills grow.

Project Loom Features

- [JEP 462: Structured Concurrency \(Second Preview\)](#): Helps developers streamline error handling and cancellation and enhance observability by introducing an API for structured concurrency. This helps promote a style of concurrent programming that can eliminate common risks arising from cancellation and shutdown – such as thread leaks and cancellation delays – and improves the observability of concurrent code.
- [JEP 464: Scoped Values \(Second Preview\)](#): Helps increase ease-of-use, comprehensibility, performance, and robustness of developers' projects by enabling the sharing of immutable data within and across threads.

Project Panama Features

- [JEP 454: Foreign Function & Memory API](#): Increases ease-of-use, flexibility, safety, and performance for developers by introducing an API to enable Java programs to interoperate with code and data outside of the Java runtime. By efficiently invoking foreign functions such as code outside the Java Virtual Machine, and by safely accessing foreign memory (i.e., memory not managed by the JVM), the new API allows Java programs to call native libraries and process native data without requiring the Java Native Interface.
- [JEP 460: Vector API \(Seventh Incubator\)](#): Enables developers to achieve performance superior to equivalent scalar computations by introducing an API to express vector

computations that reliably compile at runtime to vector instructions on supported CPU architectures.

Core Libraries & Tools Features

- [JEP 457: Class-File API \(Preview\)](#): Helps developers improve productivity by providing a standard API for parsing, generating, and transforming Java class files.
- [JEP 458: Launch Multi-File Source-Code Programs](#): Enables developers to choose whether and when to configure a build tool by enhancing the Java application launcher to enable it to run a program supplied as multiple files of Java source code.
- [JEP 461: Stream Gatherers \(Preview\)](#): Helps developers improve productivity by enhancing the Stream API to support custom intermediate operations, which will allow stream pipelines to transform data in ways that are not easily achievable with the existing built-in intermediate operations. By making stream pipelines more flexible and expressive and allowing custom intermediate operations to manipulate streams of infinite size, this feature enables developers to become more efficient in reading, writing, and maintaining Java code.

Performance Updates

- [JEP 423: Region Pinning for G1](#): Helps reduce latency by allowing some garbage collection to happen during some native library calls that would have otherwise needed to pause the collector. This is achieved by tracking which objects need to be blocked during these native library calls and “pinning” just the regions that contain these objects. This allows garbage collection to continue normally in unpinned regions, even during what would have otherwise been a blocking native library call.

The Java 22 release is the result of continuous collaboration between Oracle and other members of the worldwide Java developer community via OpenJDK and the Java Community Process (JCP). In addition to the new enhancements and features, Java 22 is supported by [Java Management Service \(JMS\)](#) – an [Oracle Cloud Infrastructure \(OCI\)](#) native service – which offers a unified console and dashboard to help organizations manage Java runtimes and applications on-premises or in any cloud. For more details on the features in Java 22, please read the Java 22 [technical blog post](#).

Supporting the Global Java Ecosystem with Innovation in the Cloud

Java delivers increased performance, efficiency, innovation, and cost savings when deployed in the cloud on OCI, which is one of the first hyperscale clouds to support Java 22. By delivering Oracle Java SE, Oracle GraalVM, and the Java SE Subscription Enterprise Performance Pack free of charge on OCI, Java 22 enables developers to build and deploy applications that run faster, better, and with optimized cost-performance.

The Oracle Java Universal SE Subscription is a pay-as-you-go offering that provides customers with best-in-class support. It includes triage support for the entire Java portfolio, entitlement to GraalVM, the Java SE Subscription Enterprise Performance Pack, access to the advanced features of the Java Management Service, and the flexibility to upgrade at the pace of their businesses. This helps IT organizations manage complexity, contain costs, and mitigate security risks.

JavaOne Returns in 2025

The flagship event for the global Java community, JavaOne, will return to the San Francisco Bay Area in 2025. Taking place from March 17-20, 2025 in Redwood Shores, California, JavaOne 2025 will give attendees the opportunity to hear about the latest Java developments and interact with Oracle's Java experts and industry luminaries. Read more via the blog post on Inside Java [here](#).

Supporting Quotes

“Within the dynamic ecosystem of the [Türkiye Java Community](#) (TJC), Oracle's enduring support fuels our momentum,” said Evren Tan, community lead, TJC. “As Java 22 unfolds, we are embracing an array of contemporary features that further enhance developer efficiency, bolstering our commitment to advancing Java technology in Türkiye. We also welcome Java developers worldwide to join us in TJC activities, showcasing the welcoming community spirit that keeps Java vibrant.”

“Java's popularity as a programming language, platform, and developer community continues to grow in Morocco and the African region,” said Badr El Hourari, founder and CIO, xHub. “With Java 22, I'm excited to see Oracle's Java team focus on innovation that will help the new developers adopt Java more quickly such as with JEP 463. By simplifying the language, the on-ramp to Java becomes easier for a whole new generation of programmers. Also, here in Morocco we continue to highlight the importance of Java at the annual Devovx Morocco conference, and we welcome all developers to attend this year's event on October 2-4 to hear from some of the most influential global voices in the Java community.”

“It amazes me to see Java 22 bundle features for all – from students to senior developers, and from adventurous Java enthusiasts to organizations looking for stability with performance,” said Mala Gupta, Java developer advocate, JetBrains. “Java is embracing a minimalist approach by continuing to reduce ceremonies around writing initial steps of code, via instance main method and implicit classes, making it easier for folks to get started learning Java. With constructor makeover (statements before super[...]), Java has proven again that it supports responsible innovation, and relaxing language constraints that existed from Java's version 1.0 without breaking any existing code is very exciting. IntelliJ IDEA 2024.1 is ready with its support for Java 22 features to enable Java developers to use them with ease.”

“The Helidon team recognized the value of virtual threads as soon as they were announced. With Helidon 4, the team rewrote the Helidon web server to natively use virtual threads, which resulted in much better performance and increased developer productivity,” said Tom Snyder, vice president of engineering, enterprise cloud native Java, Oracle. “Now with Java 22 we are seeing another performance boost for virtual threads, and these significant improvements will benefit all Helidon users.”

“Seamless integration with the latest Java innovations are helping Oracle Database deliver the world's most advanced database technologies for developing and running modern applications,” said Kuassi Mensah, director of product management, Oracle. “The Oracle Java Database Connectivity (JDBC)



drivers have been rewritten to seamlessly support Java virtual threads. The combination of Java virtual threads and the new pipelined database operations in Oracle Database 23c helps improve end-to-end scalability for Java applications.”

To learn more about Java and its global ecosystem, please visit:

- [Dev.java](#): The official portal for learning Java
- [Inside.java](#): News and views from the members of the Java Team at Oracle
- [Java YouTube](#): The official Java YouTube channel for Java learning videos

Additional Resources

- [Download Oracle JDK 22](#)
- Read the [Java 22 technical blog](#)
- Watch the [Java 22 livestream](#)
- Read about [JavaOne 2025](#)
- Learn more about [Java Management Service](#)
- Learn more about the [Oracle Java SE Universal Subscription](#)

About Oracle

Oracle offers integrated suites of applications plus secure, autonomous infrastructure in the Oracle Cloud. For more information about Oracle (NYSE: ORCL), please visit us at www.oracle.com.

Trademarks

Oracle, Java, MySQL and NetSuite are registered trademarks of Oracle Corporation. NetSuite was the first cloud company—ushering in the new era of cloud computing.

Contact Info

Drew Smith

Oracle

drew.j.smith@oracle.com

415-336-1103

###