

# JSR 399: Java SE 24

**Iris Clark**

Specification Lead

[iris.clark@oracle.com](mailto:iris.clark@oracle.com)

December 10, 2024



# JSR 399: Java SE 24

## Specification

- Latest: <https://cr.openjdk.org/~iris/se/24/latestSpec> (soon!)
- Public Review to begin January

## Reference Implementation (RI) – JDK 24

- Latest: <https://jdk.java.net/24> (build 27)
- Repository: <https://github.com/openjdk/jdk/tree/jdk24>
- Rampdown Phase 1 (RDP1)
  - Feature set frozen
  - Development limited to selected bug fixes
- 13 Integrated SE JEPs; 108 approved SE CSRs
- General Availability (GA): 2025/03/18

## Technology Compatibility Took Kit (TCK) – JCK 24

- Code freeze rapidly approaching; Stabilization fork soon thereafter

## Schedule

2024/06

Expert Group Formation

2025/01 – 2025/02

Public Review

2025/02 – 2025/03

Public Review –  
Final Approval Ballot

2025/03

Final Release

# SE JEPs in Java SE 24

## Language

- [488](#) Primitive Types in Patterns, instanceof, & switch (Second Preview)
- [494](#) Module Import Declarations (Second Preview)
- [492](#) Flexible Constructor Bodies (Third Preview)
- [495](#) Simple Source Files & Instance main Methods (Fourth Preview)

## Libraries

- [484](#) Class-File API
- [485](#) Stream Gatherers
- [499](#) Structured Concurrency (Fourth Preview)
- [487](#) Scoped Values (Fourth Preview)
- [472](#) Prepare to Restrict the Use of JNI
- [486](#) Permanently Disable the SecurityManager
- [478](#) Key Derivation Function API (Preview)
- [496](#) Quantum-Resistant Module-Lattice-Based Key Encapsulation Mechanism
- [497](#) Quantum-Resistant Module-Lattice-Based Digital Signature Algorithm

# Other notable changes in Java SE 24

## 108 Compatibility & Specification Review (CSR) Requests

<https://bugs.openjdk.org/issues/?filter=45707>

## 3 JSR Maintenance Releases

199: Java Compiler API [MR 8]

221: JDBC 4.0 API Specification [MR 4]

269: Pluggable Annotations  
Processing API [MR 18]

## 1 Removed API

`java.awt.Window.warningString` (24)  
(package private field)

## 5 Terminally Deprecated APIs Added

`java.awt.AWTPermission` (24)

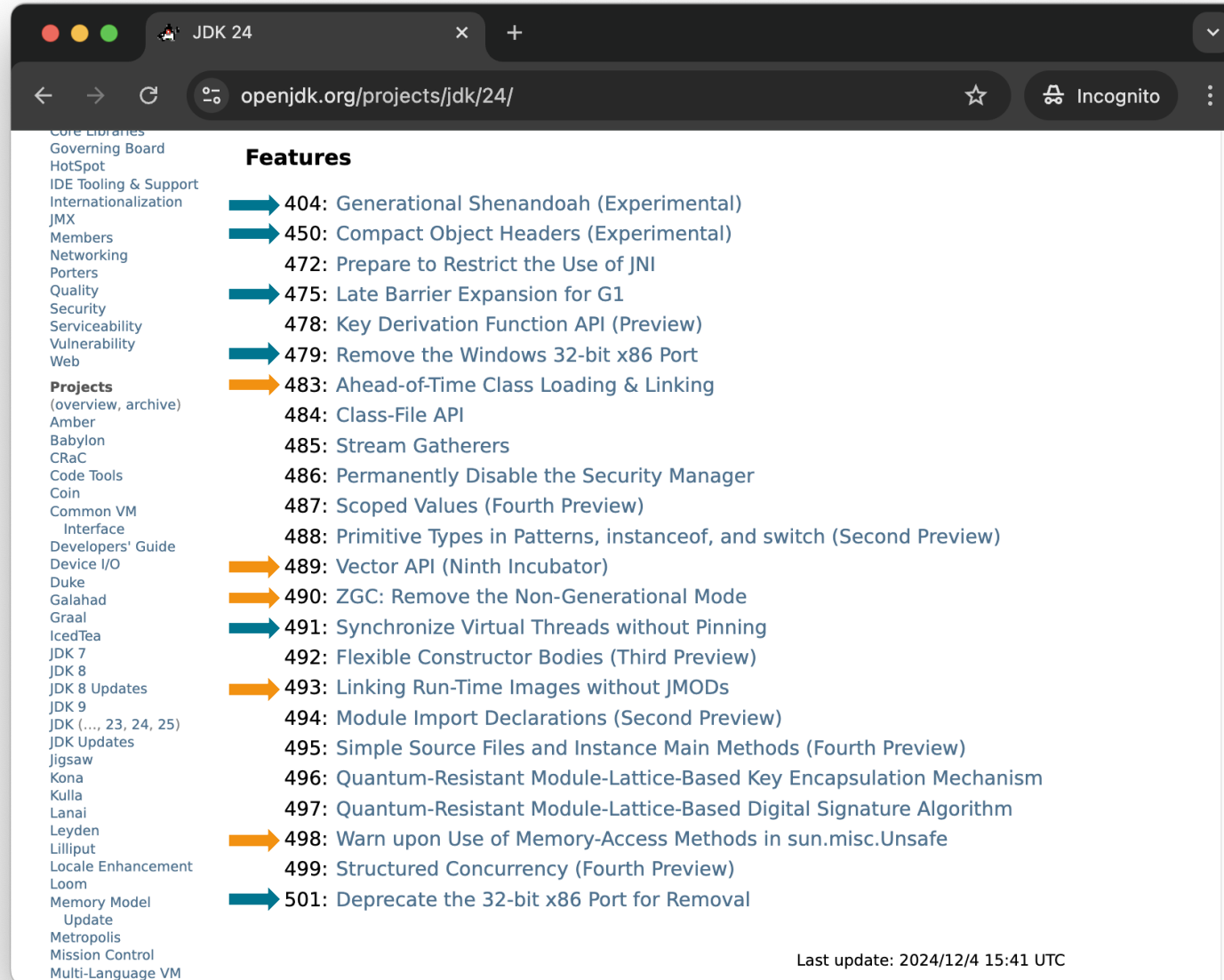
`java.awt.Window.getWarningString` (24)

`java.util.zip.ZipError` (24)

`javax.sound.sampled.AudioPermission` (24)

`javax.swing.JInternalFrame.getWarningString` (24)

# openjdk.org/projects/jdk/24



The screenshot shows a web browser window with the URL `openjdk.org/projects/jdk/24/`. The page is titled "Features" and lists 20 items, each with a colored arrow icon and a number followed by a description. The items are:

- 404: Generational Shenandoah (Experimental)
- 450: Compact Object Headers (Experimental)
- 472: Prepare to Restrict the Use of JNI
- 475: Late Barrier Expansion for G1
- 478: Key Derivation Function API (Preview)
- 479: Remove the Windows 32-bit x86 Port
- 483: Ahead-of-Time Class Loading & Linking
- 484: Class-File API
- 485: Stream Gatherers
- 486: Permanently Disable the Security Manager
- 487: Scoped Values (Fourth Preview)
- 488: Primitive Types in Patterns, instanceof, and switch (Second Preview)
- 489: Vector API (Ninth Incubator)
- 490: ZGC: Remove the Non-Generational Mode
- 491: Synchronize Virtual Threads without Pinning
- 492: Flexible Constructor Bodies (Third Preview)
- 493: Linking Run-Time Images without JMODs
- 494: Module Import Declarations (Second Preview)
- 495: Simple Source Files and Instance Main Methods (Fourth Preview)
- 496: Quantum-Resistant Module-Lattice-Based Key Encapsulation Mechanism
- 497: Quantum-Resistant Module-Lattice-Based Digital Signature Algorithm
- 498: Warn upon Use of Memory-Access Methods in `sun.misc.Unsafe`
- 499: Structured Concurrency (Fourth Preview)
- 501: Deprecate the 32-bit x86 Port for Removal

At the bottom right of the page, it says "Last update: 2024/12/4 15:41 UTC".

# Resources

- <https://openjdk.org/projects/jdk/24/spec/>
  - <https://jcp.org/en/jsr/detail?id=399>
  - JEPs: <https://bugs.openjdk.org/secure/Dashboard.jspa?selectPageId=22701>
  - CSRs: <https://bugs.openjdk.org/secure/Dashboard.jspa?selectPageId=22702>
  - <https://mail.openjdk.org/mailman/listinfo/java-se-spec-experts>
  - <https://jdk.java.net/24/>
- <https://openjdk.org/projects/jdk/25/spec/>
- <https://mail.openjdk.org>
- <https://github.com/openjdk/jdk>