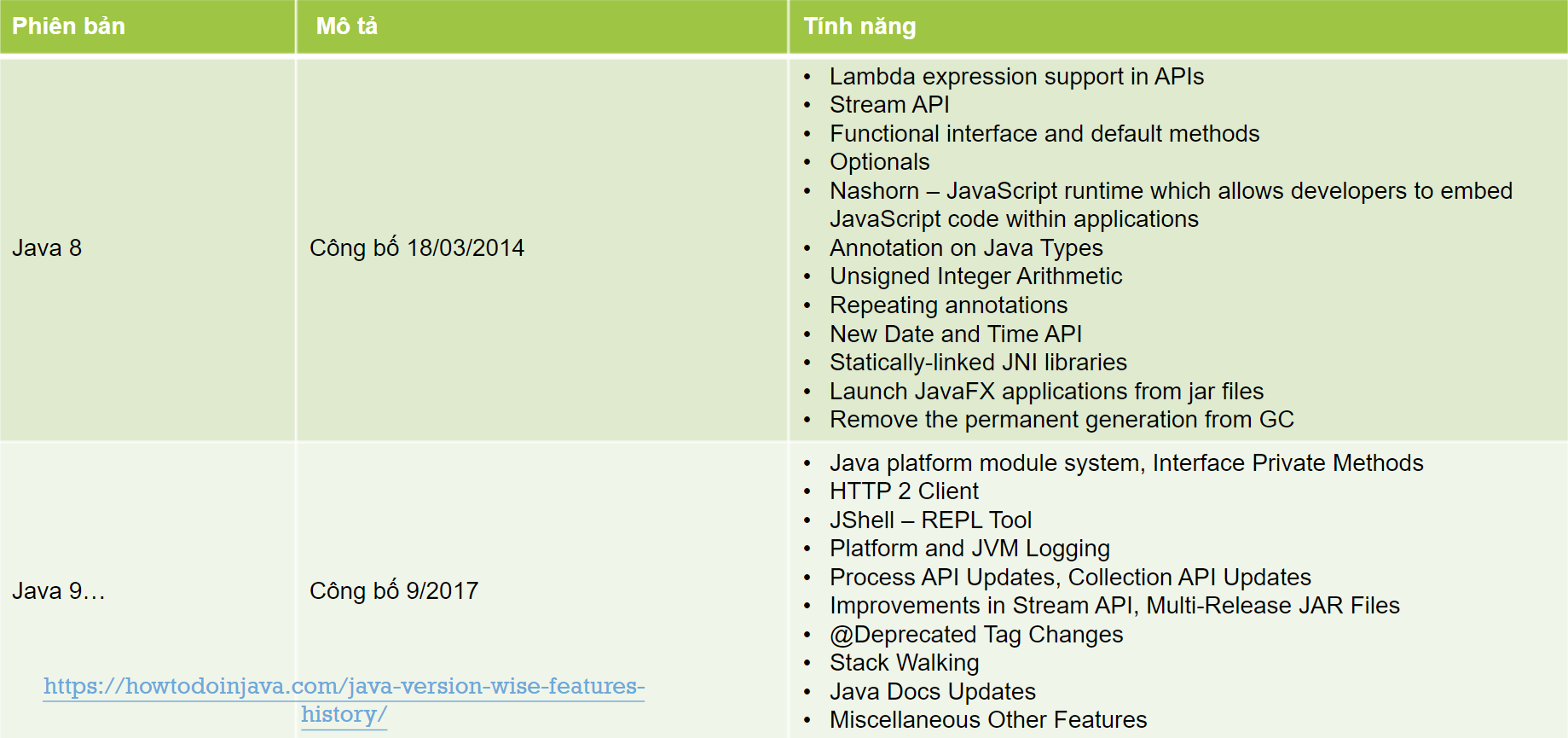
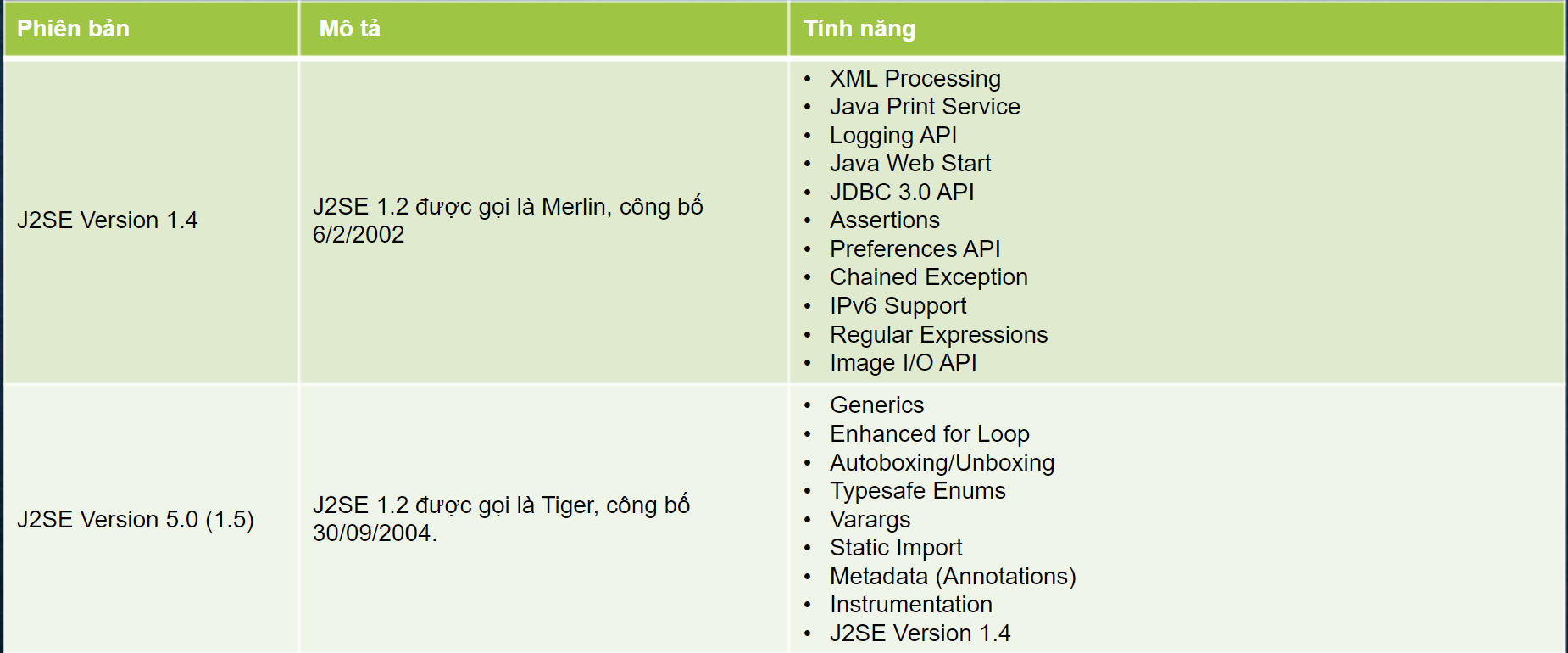
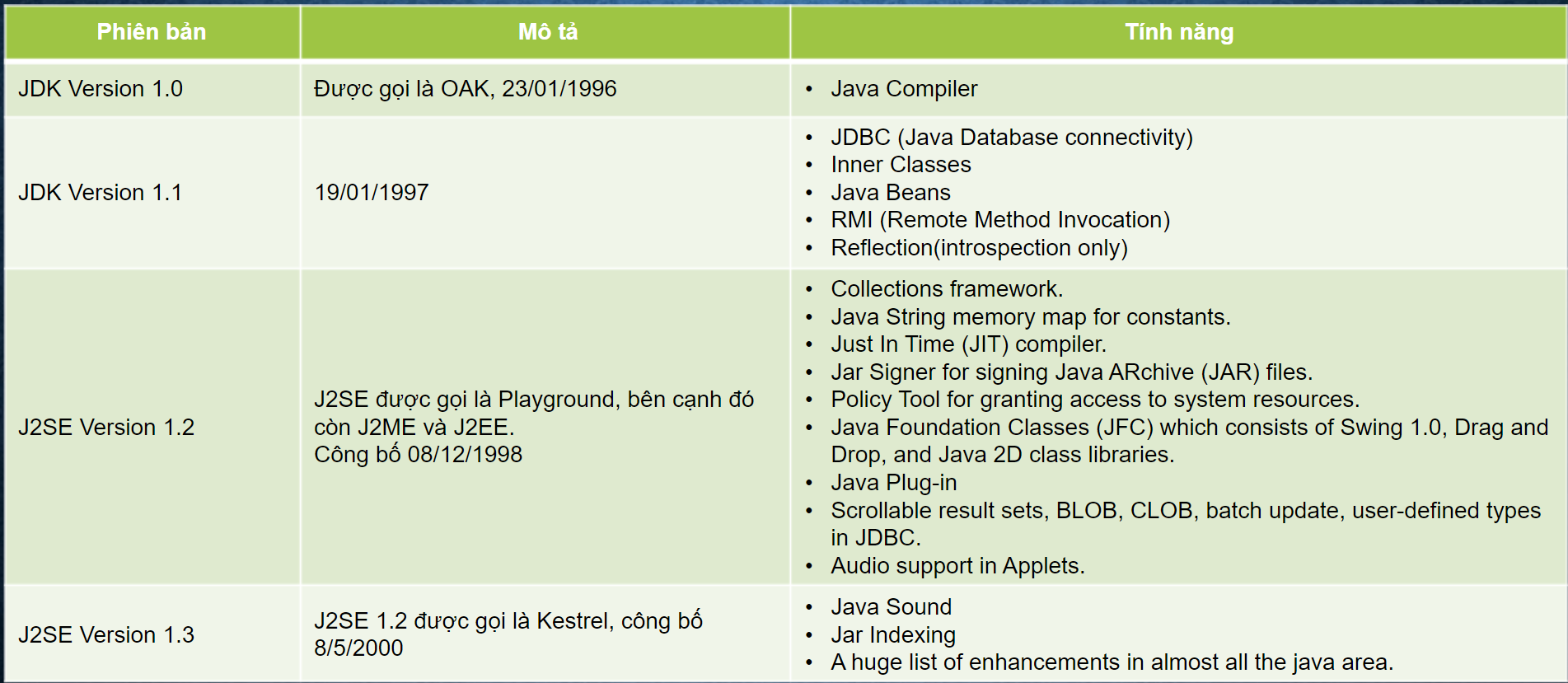
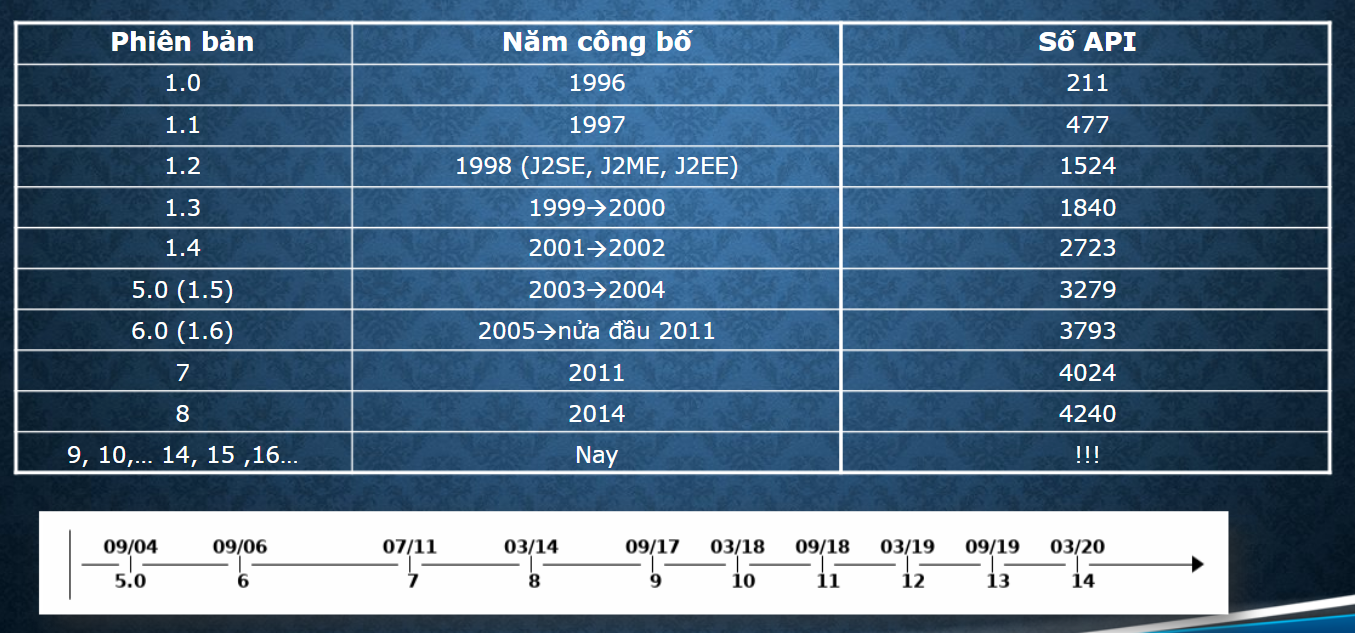
Bài 2



* JVM (Java Virtual Machine) là 1 máy ảo java - trình thông dịch của Java. Nó cung cấp môi trường để code java có thể được thực thi, chương trình Java khi biên dịch sẽ tạo ra các file \*.class chứa byte code , Các file \*.class này sẽ được JVM thực hiện chuyển byte code thành mã máy tương ứng với từng hệ điều hành và phần cứng khác nhau thực thi.
* JRE (Java Runtime Environment): JRE như là một runtime environment, nó gồm có các Java class libraries, Java class loader, và Java Vitrual Machine:
  + Java class loader chịu trách nhiệm nạp các classes và kết nối chúng với các thư viện Java (java libraries)
  + Java virtual machine(JVM) chịu trách nhiệm đảm bảo ứng dụng Java có đủ tài nguyên để chúng thực thi tốt trong thiết bị hoặc môi trường đám mây
  + JRE đảm bảo chương trình Java có thể chạy trên hầu hết các hề điều hành mà không cần sửa đổi.
* JDK (Java Development Kit) cho phép các developer tạo các chương trình Java, trong đó các chương trình có thể được JVM và JRE xử lý và chạy.
  + Định nghĩa chuyên ngành: JDK là một hệ tiêu chuẩn trong việc triển khai nền tảng Java, bao gồm các trình thông dịch dịch và thư viện lớp.
  + Định nghĩa thông thường: JDK là gói phần mềm bạn tải xuống để tạo các ứng dụng dựa trên Java.



[**Java 10 Features**](https://howtodoinjava.com/java10/java10-features/)

After Java 9 release, Java 10 came very quickly. Unlike its previous release, Java 10 does not have that many exciting features, still, it has a [few important updates](https://howtodoinjava.com/java10/java10-features/) which will change the way you code, and other future Java versions.

* [JEP 286: Local Variable Type Inference](https://howtodoinjava.com/java10/var-local-variable-type-inference/)
* JEP 322: Time-Based Release Versioning
* JEP 304: Garbage-Collector Interface
* JEP 307: Parallel Full GC for G1
* JEP 316: Heap Allocation on Alternative Memory Devices
* JEP 296: Consolidate the JDK Forest into a Single Repository
* JEP 310: Application Class-Data Sharing
* JEP 314: Additional Unicode Language-Tag Extensions
* JEP 319: Root Certificates
* JEP 317: Experimental Java-Based JIT Compiler
* JEP 312: Thread-Local Handshakes
* JEP 313: Remove the Native-Header Generation Tool
* New Added APIs and Options
* Removed APIs and Options

## [Java 11 Features (LTS)](https://howtodoinjava.com/java11/features-enhancements/)

[Java 11](https://howtodoinjava.com/java11/features-enhancements/) (released on September 2018) includes many important and useful updates. Let’s see the new features and improvements, it brings for developers and architects.

* HTTP Client API
* Launch Single-File Programs Without Compilation
* String API Changes
* Collection.toArray(IntFunction)
* Files.readString() and Files.writeString()
* Optional.isEmpty()

## [Java 12 Features](https://howtodoinjava.com/java12/new-features-enhancements/)

[Java 12](https://howtodoinjava.com/java12/new-features-enhancements/) was released on March 19, 2019. Let’s see the new features and improvements, it brings for developers and architects.

* Collectors.teeing() in Stream API
* String API Changes
* Files.mismatch(Path, Path)
* Compact Number Formatting
* Support for Unicode 11
* Switch Expressions (Preview)

## Java 13 Features

Java 13 (released on September 17, 2019) had fewer developer-specific features. Let’s see the new features and improvements, it brought for developers and architects.

* JEP 355 – Text Blocks (Preview)
* JEP 354 – Switch Expressions Enhancements (Preview)
* JEP 353 – Reimplement the Legacy Socket API
* JEP 350 – Dynamic CDS Archive
* JEP 351 – ZGC: Uncommit Unused Memory
* FileSystems.newFileSystem() Method
* DOM and SAX Factories with Namespace Support

## [Java 14 Features](https://howtodoinjava.com/java14/java14-new-features/)

[Java 14](https://howtodoinjava.com/java14/java14-new-features/) (released on March 17, 2020) is the latest version available for JDK. Let’s see the new features and improvements, it brings for developers and architects.

* [JEP 305 – Pattern Matching for instanceof (Preview)](https://howtodoinjava.com/java14/pattern-matching-instanceof/)
* [JEP 368 – Text Blocks (Second Preview)](https://howtodoinjava.com/java14/java-text-blocks/)
* [JEP 358 – Helpful NullPointerExceptions](https://howtodoinjava.com/java14/helpful-nullpointerexception/)
* [JEP 359 – Records (Preview)](https://howtodoinjava.com/java14/java-14-record-type/)
* [JEP 361 – Switch Expressions (Standard)](https://howtodoinjava.com/java14/switch-expressions/)
* JEP 343 – Packaging Tool (Incubator)
* JEP 345 – NUMA-Aware Memory Allocation for G1
* JEP 349 – JFR Event Streaming
* JEP 352 – Non-Volatile Mapped Byte Buffers
* JEP 363 – Remove the Concurrent Mark Sweep (CMS) Garbage Collector
* JEP 367 – Remove the Pack200 Tools and API
* JEP 370 – Foreign-Memory Access API (Incubator)

## [Java 15 Features](https://howtodoinjava.com/java15/java-15-new-features/)

Java 15 was released on 15th Sep’2020. It continues to support various preview features in previous JDK releases; and has also introduced some new features.

* [Sealed Classes and Interfaces](https://howtodoinjava.com/java15/sealed-classes-interfaces/) (Preview) (JEP 360)
* [EdDSA Algorithm](https://howtodoinjava.com/java15/java-eddsa-example/) (JEP 339)
* Hidden Classes (JEP 371)
* [Pattern Matching for instanceof](https://howtodoinjava.com/java14/pattern-matching-instanceof/) (Second Preview) (JEP 375)
* Removed Nashorn JavaScript Engine (JEP 372)
* Reimplement the Legacy DatagramSocket API (JEP 373)
* Records (Second Preview) (JEP 384)
* Text Blocks become a standard feature. (JEP 378)

## Java 16 Features

**Java 16** was released on 16 March 20121. It was largely a maintenance release, except it made the Java Records and Pattern matching the standard features of the Java language.

* JEP 338: Vector API (Incubator)
* JEP 347: Enable C++14 Language Features
* JEP 357: Migrate from Mercurial to Git
* JEP 369: Migrate to GitHub
* JEP 376: ZGC: Concurrent Thread-Stack Processing
* JEP 380: Unix-Domain Socket Channels
* JEP 386: Alpine Linux Port
* JEP 387: Elastic Metaspace
* JEP 388: Windows/AArch64 Port
* JEP 389: Foreign Linker API (Incubator)
* JEP 390: Warnings for Value-Based Classes
* JEP 392: Packaging Tool
* JEP 393: Foreign-Memory Access API (Third Incubator)
* JEP 394: Pattern Matching for instanceof
* JEP 395: Records
* JEP 396: Strongly Encapsulate JDK Internals by Default
* JEP 397: Sealed Classes (Second Preview)

## [Java 17 Features (LTS)](https://howtodoinjava.com/java17/new-features/)

**Java 17** was released on September 14, 2021. Java 17 is an LTS (**Long Term Support**) release, like Java 11 and Java 8. [Spring 6 and Spring boot 3](https://www.infoq.com/news/2021/09/spring-6-spring-boot-3-overhaul/) will have first-class support for Java 17. So it is a good idea to plan for upgrading to Java 17.

The below-listed 14 JEPs are part of Java 17.

* ([JEP-306](https://openjdk.java.net/jeps/306)) Restore Always-Strict Floating-Point Semantics
* ([JEP-356](https://openjdk.java.net/jeps/356)) Enhanced Pseudo-Random Number Generators
* ([JEP-382](https://openjdk.java.net/jeps/382)) New macOS Rendering Pipeline
* ([JEP-391](https://openjdk.java.net/jeps/391)) macOS/AArch64 Port
* ([JEP-398](https://openjdk.java.net/jeps/398)) Deprecate the Applet API for Removal
* ([JEP-403](https://openjdk.java.net/jeps/403)) Strongly Encapsulate JDK Internals
* ([JEP-406](https://openjdk.java.net/jeps/406)) Pattern Matching for switch (Preview)
* ([JEP-407](https://openjdk.java.net/jeps/407)) Remove RMI Activation
* ([JEP-409](https://openjdk.java.net/jeps/409)) Sealed Classes
* ([JEP-410](https://openjdk.java.net/jeps/410)) Remove the Experimental AOT and JIT Compiler
* ([JEP-411](https://openjdk.java.net/jeps/411)) Deprecate the Security Manager for Removal
* ([JEP-412](https://openjdk.java.net/jeps/412)) Foreign Function & Memory API (Incubator)
* ([JEP-414](https://openjdk.java.net/jeps/414)) Vector API (Second Incubator)
* ([JEP-415](https://openjdk.java.net/jeps/415)) Context-Specific Deserialization Filters

## Java 18 Features

**Java 18** GA was released on 22 March 2022. It has nine new developer features, including a simple web server and another preview of pattern matching for switch.

The below-listed 14 JEPs are part of Java 18.

* [JEP-400](https://openjdk.java.net/jeps/400): UTF-8 by Default
* [JEP-408](https://openjdk.java.net/jeps/408): Simple Web Server
* [JEP-413](https://openjdk.java.net/jeps/408): Code Snippets in Java API Documentation
* [JEP-416](https://openjdk.java.net/jeps/416): Reimplement Core Reflection with Method Handles
* [JEP-417](https://openjdk.java.net/jeps/417): Vector API (Third Incubator)
* [JEP-418](https://openjdk.java.net/jeps/418): Internet-Address Resolution SPI
* [JEP-419](https://openjdk.java.net/jeps/419): Foreign Function & Memory API (Second Incubator)
* [JEP-420](https://openjdk.java.net/jeps/420): Pattern Matching for switch (Second Preview)
* [JEP-421](https://openjdk.java.net/jeps/421): Deprecate Finalization for Removal