**API Enhancements and New Features:**

1. **HTTP Client (Standard) API**:
2. **Local-Variable Syntax for Lambda Parameters**:
3. **Nested Based Access Control**:
4. **Optional.isEmpty()**:
5. **String Methods**:
   * New methods in the String class: lines(), strip(), stripLeading(), stripTrailing(), repeat(int), and isBlank().
6. **Files.readString() and Files.writeString()**:

**Memory Management and Performance Improvements:**

1. **ZGC (Z Garbage Collector)**:
   * A scalable low-latency garbage collector designed for applications requiring large heaps with low pause times.
   * To use ZGC, start the JVM with the -XX:+UseZGC option.
2. **Epsilon Garbage Collector**:
   * A no-op garbage collector that handles memory allocation but does not reclaim memory. Useful for performance testing and memory pressure testing.
   * To use Epsilon GC, start the JVM with the -XX:+UseEpsilonGC option.
3. **Heap Allocation on Alternative Memory Devices**:
   * Support for heap allocation on non-volatile memory (NVM) devices.
4. **Dynamic Class-File Constants (JEP 309)**:
   * Allows the JVM to load and process constant pool entries at runtime, enabling more dynamic and flexible constant handling.
5. **Flight Recorder**:
   * Java Flight Recorder (JFR) is a tool for collecting diagnostic and profiling data about a running Java application with low overhead.
   * It can be enabled with the -XX:StartFlightRecording option.

**Deprecations and Removals:**

1. **Removal of Java EE and CORBA Modules**:
   * The Java EE (javax.xml.bind, javax.xml.ws, etc.) and CORBA modules were removed from the JDK.
2. **Nashorn JavaScript Engine Deprecated**:
   * The Nashorn JavaScript engine has been deprecated in favor of other JavaScript engines like GraalVM.
3. **Pack200 Tools and API Deprecated**:
   * The pack200 and unpack200 tools and their associated APIs have been deprecated.