New API since JDK 11

Show API added in: ☑ 12 ☑ 13 ☑ 14 ☑ 15 ☑ 16 ☑ 17 ☑ 18 ☑ 19 ☑ 20 ☑ 21

Contents

New Modules

Module ⬍	Added in ⬍	Description
jdk.jpackage	16	Defines the Java Packaging tool, jpackage.
jdk.nio.mapmode	14	Defines JDK-specific file mapping modes.

New Packages

Package ⬍	Added in ⬍	Description
java.lang.constant	12	Classes and interfaces to represent *nominal descriptors* for run-time entities such as classes or method handles, and classfile entities such as constant pool entries or `invokedynamic` call sites.
java.lang.runtime	14	The `java.lang.runtime` package provides low-level runtime support for the Java language.
java.util.random	17	This package contains classes and interfaces that support a generic API for random number generation.

New Interfaces

Interface ⬍	Added in ⬍	Description
com.sun.net.httpserver.Request	18	A view of the immutable request state of an HTTP exchange.
com.sun.source.doctree.EscapeTree	21	A tree node for a character represented by an escape sequence.
com.sun.source.doctree.SnippetTree	18	A tree node for an `@snippet` inline tag.
com.sun.source.doctree.SpecTree	20	A tree node for an `@spec` block tag.
com.sun.source.doctree.SystemPropertyTree	12	A tree node for an `@systemProperty` inline tag.
com.sun.source.tree.AnyPatternTree^{PREVIEW}	21	A tree node for a binding pattern that matches a pattern with a variable of any name and a type of the match candidate; an unnamed pattern.
com.sun.source.tree.BindingPatternTree	16	A binding pattern tree
com.sun.source.tree.CaseLabelTree	21	A marker interface for `Trees` that may be used as `CaseTree` labels.
com.sun.source.tree.ConstantCaseLabelTree	21	A case label element that refers to a constant expression
com.sun.source.tree.DeconstructionPatternTree	21	A deconstruction pattern tree.
com.sun.source.tree.DefaultCaseLabelTree	21	A case label that marks `default` in `case null, default`.
com.sun.source.tree.PatternCaseLabelTree	21	A case label element that refers to an expression
com.sun.source.tree.PatternTree	16	A tree node used as the base class for the different kinds of patterns.

com.sun.source.tree.SwitchExpressionTree	12	A tree node for a `switch` expression.
com.sun.source.tree.YieldTree	13	A tree node for a `yield` statement.
com.sun.source.util.ParameterNameProvider	13	A provider for parameter names when the parameter names are not determined from a reliable source, like a classfile.
java.lang.constant.ClassDesc	12	A nominal descriptor for a `Class` constant.
java.lang.constant.Constable	12	Represents a type which is *constable*.
java.lang.constant.ConstantDesc	12	A nominal descriptor for a loadable constant value, as defined in JVMS 4.4⧉.
java.lang.constant.DirectMethodHandleDesc	12	A nominal descriptor for a direct `MethodHandle`.
java.lang.constant.MethodHandleDesc	12	A nominal descriptor for a `MethodHandle` constant.
java.lang.constant.MethodTypeDesc	12	A nominal descriptor for a `MethodType` constant.
java.lang.constant.ModuleDesc	21	A nominal descriptor for a `Module` constant.
java.lang.constant.PackageDesc	21	A nominal descriptor for a `Package` constant.
java.lang.foreign.AddressLayout^{PREVIEW}	19	A value layout used to model the address of some region of memory.
java.lang.foreign.Arena^{PREVIEW}	20	An arena controls the lifecycle of native memory segments, providing both flexible allocation and timely deallocation.
java.lang.foreign.FunctionDescriptor^{PREVIEW}	19	A function descriptor models the signature of a foreign function.
java.lang.foreign.GroupLayout^{PREVIEW}	19	A compound layout that is an aggregation of multiple, heterogeneous *member layouts*.
java.lang.foreign.Linker^{PREVIEW}	19	A linker provides access to foreign functions from Java code, and access to Java code from foreign functions.
java.lang.foreign.Linker.Option^{PREVIEW}	20	A linker option is used to provide additional parameters to a linkage request.
java.lang.foreign.MemoryLayout^{PREVIEW}	19	A memory layout describes the contents of a memory segment.
java.lang.foreign.MemoryLayout.PathElement^{PREVIEW}	19	An element in a *layout path*.
java.lang.foreign.MemorySegment^{PREVIEW}	19	A memory segment provides access to a contiguous region of memory.
java.lang.foreign.PaddingLayout^{PREVIEW}	20	A padding layout.
java.lang.foreign.SequenceLayout^{PREVIEW}	19	A compound layout that denotes a homogeneous repetition of a given *element layout*.
java.lang.foreign.StructLayout^{PREVIEW}	20	A group layout whose member layouts are laid out one after the other.
java.lang.foreign.UnionLayout^{PREVIEW}	20	A group layout whose member layouts are laid out at the same starting offset.
java.lang.foreign.ValueLayout^{PREVIEW}	19	A layout that models values of basic data types.
java.lang.foreign.ValueLayout.OfBoolean^{PREVIEW}	19	A value layout whose carrier is `boolean.class`.
java.lang.foreign.ValueLayout.OfByte^{PREVIEW}	19	A value layout whose carrier is `byte.class`.
java.lang.foreign.ValueLayout.OfChar^{PREVIEW}	19	A value layout whose carrier is `char.class`.
java.lang.foreign.ValueLayout.OfDouble^{PREVIEW}	19	A value layout whose carrier is `double.class`.
java.lang.foreign.ValueLayout.OfFloat^{PREVIEW}	19	A value layout whose carrier is `float.class`.
java.lang.foreign.ValueLayout.OfInt^{PREVIEW}	19	A value layout whose carrier is `int.class`.
java.lang.foreign.ValueLayout.OfLong^{PREVIEW}	19	A value layout whose carrier is `long.class`.
java.lang.foreign.ValueLayout.OfShort^{PREVIEW}	19	A value layout whose carrier is `short.class`.
java.lang.invoke.TypeDescriptor	12	An entity that has a type descriptor.
java.lang.invoke.TypeDescriptor.OfField	12	An entity that has a field type descriptor.

java.lang.invoke.TypeDescriptor.OfMethod	12	An entity that has a method type descriptor Method descriptors conforming to JVMS 4.3.3 can be described nominally via `MethodType::describeConstable`; otherwise they cannot be described nominally.
java.lang.StringTemplate[PREVIEW]	21	`StringTemplate`[PREVIEW] is the run-time representation of a string template or text block template in a template expression.
java.lang.StringTemplate.Processor[PREVIEW]	21	This interface describes the methods provided by a generalized string template processor.
java.lang.StringTemplate.Processor.Linkage[PREVIEW]	21	Built-in policies using this additional interface have the flexibility to specialize the composition of the templated string by returning a customized `MethodHandle` from `linkage`[PREVIEW].
java.lang.Thread.Builder	21	A builder for `Thread` and `ThreadFactory` objects.
java.lang.Thread.Builder.OfPlatform	21	A builder for creating a platform `Thread` or `ThreadFactory` that creates platform threads.
java.lang.Thread.Builder.OfVirtual	21	A builder for creating a virtual `Thread` or `ThreadFactory` that creates virtual threads.
java.net.spi.InetAddressResolver	18	This interface defines operations for looking up host names and IP addresses.
java.net.spi.InetAddressResolverProvider.Configuration	18	A `Configuration` object is supplied to the `InetAddressResolverProvider.get(Configuration)` method when setting the system-wide resolver.
java.security.interfaces.EdECKey	15	An interface for an elliptic curve public/private key as defined by RFC 8032: Edwards-Curve Digital Signature Algorithm (EdDSA).
java.security.interfaces.EdECPrivateKey	15	An interface for an elliptic curve private key as defined by RFC 8032: Edwards-Curve Digital Signature Algorithm (EdDSA).
java.security.interfaces.EdECPublicKey	15	An interface for an elliptic curve public key as defined by RFC 8032: Edwards-Curve Digital Signature Algorithm (EdDSA).
java.time.InstantSource	17	Provides access to the current instant.
java.util.concurrent.StructuredTaskScope.Subtask[PREVIEW]	21	Represents a subtask forked with `StructuredTaskScope.fork(Callable)`[PREVIEW].
java.util.random.RandomGenerator	17	The `RandomGenerator` interface is designed to provide a common protocol for objects that generate random or (more typically) pseudorandom sequences of numbers (or Boolean values).
java.util.SequencedCollection	21	A collection that has a well-defined encounter order, that supports operations at both ends, and that is reversible.
java.util.SequencedMap	21	A Map that has a well-defined encounter order, that supports operations at both ends, and that is reversible.
java.util.SequencedSet	21	A collection that is both a `SequencedCollection` and a `Set`.
java.util.stream.DoubleStream.DoubleMapMultiConsume	16	Represents an operation that accepts a `double`-valued argument and a DoubleConsumer, and returns no result.
java.util.stream.IntStream.IntMapMultiConsumer	16	Represents an operation that accepts an `int`-valued argument and an IntConsumer, and returns no result.
java.util.stream.LongStream.LongMapMultiConsumer	16	Represents an operation that accepts a `long`-valued argument and a LongConsumer, and returns no result.
javax.crypto.KEMSpi	21	This class defines the Service Provider Interface (SPI) for the `KEM` class.
javax.crypto.KEMSpi.DecapsulatorSpi	21	The KEM decapsulator implementation, generated by `KEMSpi.engineNewDecapsulator(java.security.Private java.security.spec.AlgorithmParameterSpec)` on the KEM receiver side.

javax.crypto.KEMSpi.EncapsulatorSpi	21	The KEM encapsulator implementation, generated by `KEMSpi.engineNewEncapsulator(java.security.PublicK` `java.security.spec.AlgorithmParameterSpec,` `java.security.SecureRandom)` on the KEM sender side.
javax.lang.model.element.RecordComponentElement	16	Represents a record component.
jdk.jfr.consumer.EventStream	14	Represents a stream of events.
jdk.jshell.JShellConsole	21	An interface providing functionality for `Console` in the user's snippet.

New Classes

Class ⇕	Added in ⇕	Description
com.sun.net.httpserver.HttpHandlers	18	Implementations of `HttpHandler` that implement various useful handlers, such as a static response handler, or a conditional handler that complements one handler with another.
com.sun.net.httpserver.SimpleFileServer	18	A simple HTTP file server and its components (intended for testing, development and debugging purposes only).
java.lang.constant.ConstantDescs	12	Predefined values of nominal descriptor for common constants, including descriptors for primitive class types and other common platform types, and descriptors for method handles for standard bootstrap methods.
java.lang.constant.DynamicCallSiteDesc	12	A nominal descriptor for an `invokedynamic` call site.
java.lang.constant.DynamicConstantDesc	12	A nominal descriptor for a dynamic constant (one described in the constant pool with `Constant_Dynamic_info`.)
java.lang.Enum.EnumDesc	12	A nominal descriptor for an `enum` constant.
java.lang.invoke.VarHandle.VarHandleDesc	12	A nominal descriptor for a `VarHandle` constant.
java.lang.Record	16	This is the common base class of all Java language record classes.
java.lang.reflect.RecordComponent	16	A `RecordComponent` provides information about, and dynamic access to, a component of a record class.
java.lang.runtime.ObjectMethods	16	Bootstrap methods for state-driven implementations of core methods, including `Object.equals(Object)`, `Object.hashCode()`, and `Object.toString()`.
java.lang.runtime.SwitchBootstraps	21	Bootstrap methods for linking `invokedynamic` call sites that implement the selection functionality of the `switch` statement.
java.lang.runtime.TemplateRuntimePREVIEW	21	Manages string template bootstrap methods.
java.lang.ScopedValuePREVIEW	21	A value that may be safely and efficiently shared to methods without using method parameters.
java.lang.ScopedValue.CarrierPREVIEW	21	A mapping of scoped values, as *keys*, to values.
java.net.spi.InetAddressResolver.LookupPolicy	18	A `LookupPolicy` object describes characteristics that can be applied to a lookup operation.
java.net.spi.InetAddressResolverProvider	18	Service-provider class for InetAddress resolvers.
java.net.UnixDomainSocketAddress	16	A Unix domain socket address.
java.security.spec.EdDSAParameterSpec	15	A class used to specify EdDSA signature and verification parameters.
java.security.spec.EdECPoint	15	An elliptic curve point used to specify keys as defined by RFC 8032: Edwards-Curve Digital Signature Algorithm (EdDSA).
java.security.spec.EdECPrivateKeySpec	15	A class representing elliptic curve private keys as defined in RFC 8032: Edwards-Curve Digital Signature Algorithm (EdDSA), including the curve and other algorithm parameters.

java.security.spec.EdECPublicKeySpec	15	A class representing elliptic curve public keys as defined in RFC 8032: Edwards-Curve Digital Signature Algorithm (EdDSA), including the curve and other algorithm parameters.
java.text.CompactNumberFormat	12	CompactNumberFormat is a concrete subclass of NumberFormat that formats a decimal number in its compact form.
java.util.concurrent.StructuredTaskScope[PREVIEW]	21	A basic API for *structured concurrency*.
java.util.concurrent.StructuredTaskScope.ShutdownOnFa	21	A StructuredTaskScope that captures the exception of the first subtask to fail[PREVIEW].
java.util.concurrent.StructuredTaskScope.ShutdownOnSu	21	A StructuredTaskScope that captures the result of the first subtask to complete successfully[PREVIEW].
java.util.FormatProcessor[PREVIEW]	21	This StringTemplate.Processor[PREVIEW] constructs a String result using Formatter specifications and values found in the StringTemplate[PREVIEW].
java.util.HexFormat	17	HexFormat converts between bytes and chars and hex-encoded strings which may include additional formatting markup such as prefixes, suffixes, and delimiters.
java.util.random.RandomGeneratorFactory	17	This is a factory class for generating multiple random number generators of a specific algorithm.
javax.crypto.KEM	21	This class provides the functionality of a Key Encapsulation Mechanism (KEM).
javax.crypto.KEM.Decapsulator	21	A decapsulator, generated by KEM.newDecapsulator(java.security.PrivateKey) on the KEM receiver side.
javax.crypto.KEM.Encapsulated	21	This class specifies the return value of the encapsulate method of a Key Encapsulation Mechanism (KEM), which includes the shared secret (as a SecretKey), the key encapsulation message, and optional parameters.
javax.crypto.KEM.Encapsulator	21	An encapsulator, generated by KEM.newEncapsulator(java.security.PublicKey) on the KEM sender side.
javax.lang.model.util.AbstractAnnotationValueVisitor14	14	A skeletal visitor for annotation values with default behavior appropriate for source version RELEASE_14.
javax.lang.model.util.AbstractElementVisitor14	16	A skeletal visitor of program elements with default behavior appropriate for the RELEASE_14 source version.
javax.lang.model.util.AbstractTypeVisitor14	14	A skeletal visitor of types with default behavior appropriate for the RELEASE_14 source version.
javax.lang.model.util.ElementKindVisitor14	16	A visitor of program elements based on their kind with default behavior appropriate for the RELEASE_14 source version.
javax.lang.model.util.ElementScanner14	16	A scanning visitor of program elements with default behavior appropriate for the RELEASE_14 source version.
javax.lang.model.util.SimpleAnnotationValueVisitor14	14	A simple visitor for annotation values with default behavior appropriate for source version RELEASE_14.
javax.lang.model.util.SimpleElementVisitor14	16	A simple visitor of program elements with default behavior appropriate for the RELEASE_14 source version.
javax.lang.model.util.SimpleTypeVisitor14	14	A simple visitor of types with default behavior appropriate for source version RELEASE_14.
javax.lang.model.util.TypeKindVisitor14	14	A visitor of types based on their kind with default behavior appropriate for source version RELEASE_14.
javax.naming.ldap.spi.LdapDnsProvider	12	Service-provider class for DNS lookups when performing LDAP operations.
javax.naming.ldap.spi.LdapDnsProviderResult	12	The result of a DNS lookup for an LDAP URL.

javax.xml.crypto.dsig.spec.RSAPSSParameterSpec	17	Parameters for the XML Signature RSASSA-PSS Algorithm⬚.
jdk.jfr.consumer.MetadataEvent	16	Event that contains information about event types and configurations.
jdk.jfr.consumer.RecordingStream	14	A recording stream produces events from the current JVM (Java Virtual Machine).
jdk.management.jfr.RemoteRecordingStream	16	An implementation of an `EventStream` that can serialize events over the network using an `MBeanServerConnection`.
jdk.nio.mapmode.ExtendedMapMode	14	JDK-specific map modes.

New Enum Classes

Enum Class ⇕	Added in ⇕	Description
com.sun.management.HotSpotDiagnosticMXBean.Thread	21	Thread dump format.
com.sun.net.httpserver.SimpleFileServer.OutputLevel	18	Describes the log message output level produced by the server when processing exchanges.
com.sun.source.tree.CaseTree.CaseKind	12	The syntactic form of this case: STATEMENT: `case <expression>: <statements>` RULE: `case <expression> -> <expression>/<statement>`
java.lang.constant.DirectMethodHandleDesc.Kind	12	Kinds of method handles that can be described with DirectMethodHandleDesc.
java.lang.invoke.MethodHandles.Lookup.ClassOption	15	The set of class options that specify whether a hidden class created by `Lookup::defineHiddenClass` method is dynamically added as a new member to the nest of a lookup class and/or whether a hidden class has a strong relationship with the class loader marked as its defining loader.
java.lang.reflect.AccessFlag	20	Represents a JVM access or module-related flag on a runtime member, such as a class, field, or method.
java.lang.reflect.AccessFlag.Location	20	A location within a class file where flags can be applied.
java.lang.reflect.ClassFileFormatVersion	20	Class file format versions of the Java virtual machine.
java.text.NumberFormat.Style	12	A number format style.
java.util.concurrent.Future.State	19	Represents the computation state.
java.util.concurrent.StructuredTaskScope.Subtask.State	21	Represents the state of a subtask.
jdk.jshell.SourceCodeAnalysis.Attribute	19	A span attribute which can be used to derive a coloring.

New Exception Classes

Exception Class ⇕	Added in ⇕	Description
com.sun.jdi.OpaqueFrameException	19	Thrown to indicate an operation could not be performed on a frame.
java.lang.MatchException	21	Thrown to indicate an unexpected failure in pattern matching.
java.lang.WrongThreadException	19	Thrown to indicate that a method has been called on the wrong thread.
java.util.concurrent.StructureViolationException[PREVIEW]	21	Thrown when a structure violation is detected.
javax.crypto.DecapsulateException	21	An exception that is thrown by the `KEM.Decapsulator.decapsulate(byte[])` method to denote an error during decapsulation.

New Record Classes

Record Class ⇕	Added in ⇕	Description
jdk.jshell.SourceCodeAnalysis.Highlight	19	Assigns attributes usable for coloring to spans inside a snippet.

| jdk.net.UnixDomainPrincipal | 16 | Represents the credentials of a peer connected to a Unix domain socket. |

New Annotation Interfaces

Annotation Interface ⇕	Added in ⇕	Description
java.io.Serial	14	Indicates that an annotated field or method is part of the serialization mechanism defined by the *Java Object Serialization Specification*.

New Fields

Field ⇕	Added in ⇕	Description
java.lang.Character.UnicodeBlock.ARABIC_EXTENDED_B	19	Constant for the "Arabic Extended-B" Unicode character block.
java.lang.Character.UnicodeBlock.ARABIC_EXTENDED_C	20	Constant for the "Arabic Extended-C" Unicode character block.
java.lang.Character.UnicodeBlock.CHESS_SYMBOLS	12	Constant for the "Chess Symbols" Unicode character block.
java.lang.Character.UnicodeBlock.CHORASMIAN	15	Constant for the "Chorasmian" Unicode character block.
java.lang.Character.UnicodeBlock.CJK_UNIFIED_IDEOGRAI	15	Constant for the "CJK Unified Ideographs Extension G" Unicode character block.
java.lang.Character.UnicodeBlock.CJK_UNIFIED_IDEOGRAI	20	Constant for the "CJK Unified Ideographs Extension H" Unicode character block.
java.lang.Character.UnicodeBlock.CYPRO_MINOAN	19	Constant for the "Cypro-Minoan" Unicode character block.
java.lang.Character.UnicodeBlock.CYRILLIC_EXTENDED_D	20	Constant for the "Cyrillic Extended-D" Unicode character block.
java.lang.Character.UnicodeBlock.DEVANAGARI_EXTENDE	20	Constant for the "Devanagari Extended-A" Unicode character block.
java.lang.Character.UnicodeBlock.DIVES_AKURU	15	Constant for the "Dives Akuru" Unicode character block.
java.lang.Character.UnicodeBlock.DOGRA	12	Constant for the "Dogra" Unicode character block.
java.lang.Character.UnicodeBlock.EGYPTIAN_HIEROGLYPI	13	Constant for the "Egyptian Hieroglyph Format Controls" Unicode character block.
java.lang.Character.UnicodeBlock.ELYMAIC	13	Constant for the "Elymaic" Unicode character block.
java.lang.Character.UnicodeBlock.ETHIOPIC_EXTENDED_E	19	Constant for the "Ethiopic Extended-B" Unicode character block.
java.lang.Character.UnicodeBlock.GEORGIAN_EXTENDED	12	Constant for the "Georgian Extended" Unicode character block.
java.lang.Character.UnicodeBlock.GUNJALA_GONDI	12	Constant for the "Gunjala Gondi" Unicode character block.
java.lang.Character.UnicodeBlock.HANIFI_ROHINGYA	12	Constant for the "Hanifi Rohingya" Unicode character block.
java.lang.Character.UnicodeBlock.INDIC_SIYAQ_NUMBERS	12	Constant for the "Indic Siyaq Numbers" Unicode character block.
java.lang.Character.UnicodeBlock.KAKTOVIK_NUMERALS	20	Constant for the "Kaktovik Numerals" Unicode character block.
java.lang.Character.UnicodeBlock.KANA_EXTENDED_B	19	Constant for the "Kana Extended-B" Unicode character block.
java.lang.Character.UnicodeBlock.KAWI	20	Constant for the "Kawi" Unicode character block.
java.lang.Character.UnicodeBlock.KHITAN_SMALL_SCRIPT	15	Constant for the "Khitan Small Script" Unicode character block.
java.lang.Character.UnicodeBlock.LATIN_EXTENDED_F	19	Constant for the "Latin Extended-F" Unicode character block.

java.lang.Character.UnicodeBlock.LATIN_EXTENDED_G	19	Constant for the "Latin Extended-G" Unicode character block.
java.lang.Character.UnicodeBlock.LISU_SUPPLEMENT	15	Constant for the "Lisu Supplement" Unicode character block.
java.lang.Character.UnicodeBlock.MAKASAR	12	Constant for the "Makasar" Unicode character block.
java.lang.Character.UnicodeBlock.MAYAN_NUMERALS	12	Constant for the "Mayan Numerals" Unicode character block.
java.lang.Character.UnicodeBlock.MEDEFAIDRIN	12	Constant for the "Medefaidrin" Unicode character block.
java.lang.Character.UnicodeBlock.NAG_MUNDARI	20	Constant for the "Nag Mundari" Unicode character block.
java.lang.Character.UnicodeBlock.NANDINAGARI	13	Constant for the "Nandinagari" Unicode character block.
java.lang.Character.UnicodeBlock.NYIAKENG_PUACHUE_H	13	Constant for the "Nyiakeng Puachue Hmong" Unicode character block.
java.lang.Character.UnicodeBlock.OLD_SOGDIAN	12	Constant for the "Old Sogdian" Unicode character block.
java.lang.Character.UnicodeBlock.OLD_UYGHUR	19	Constant for the "Old Uyghur" Unicode character block.
java.lang.Character.UnicodeBlock.OTTOMAN_SIYAQ_NUM	13	Constant for the "Ottoman Siyaq Numbers" Unicode character block.
java.lang.Character.UnicodeBlock.SMALL_KANA_EXTENSI	13	Constant for the "Small Kana Extension" Unicode character block.
java.lang.Character.UnicodeBlock.SOGDIAN	12	Constant for the "Sogdian" Unicode character block.
java.lang.Character.UnicodeBlock.SYMBOLS_AND_PICTOG	13	Constant for the "Symbols and Pictographs Extended-A" Unicode character block.
java.lang.Character.UnicodeBlock.SYMBOLS_FOR_LEGACY	15	Constant for the "Symbols for Legacy Computing" Unicode character block.
java.lang.Character.UnicodeBlock.TAMIL_SUPPLEMENT	13	Constant for the "Tamil Supplement" Unicode character block.
java.lang.Character.UnicodeBlock.TANGSA	19	Constant for the "Tangsa" Unicode character block.
java.lang.Character.UnicodeBlock.TANGUT_SUPPLEMENT	15	Constant for the "Tangut Supplement" Unicode character block.
java.lang.Character.UnicodeBlock.TOTO	19	Constant for the "Toto" Unicode character block.
java.lang.Character.UnicodeBlock.UNIFIED_CANADIAN_AE	19	Constant for the "Unified Canadian Aboriginal Syllabics Extended-A" Unicode character block.
java.lang.Character.UnicodeBlock.VITHKUQI	19	Constant for the "Vithkuqi" Unicode character block.
java.lang.Character.UnicodeBlock.WANCHO	13	Constant for the "Wancho" Unicode character block.
java.lang.Character.UnicodeBlock.YEZIDI	15	Constant for the "Yezidi" Unicode character block.
java.lang.Character.UnicodeBlock.ZNAMENNY_MUSICAL_I	19	Constant for the "Znamenny Musical Notation" Unicode character block.
java.lang.constant.ConstantDescs.BSM_CLASS_DATA	21	`MethodHandleDesc` representing `MethodHandles.classData`
java.lang.constant.ConstantDescs.BSM_CLASS_DATA_AT	21	`MethodHandleDesc` representing `MethodHandles.classDataAt`
java.lang.constant.ConstantDescs.BSM_EXPLICIT_CAST	15	`MethodHandleDesc` representing `ConstantBootstraps.explicitCast`
java.lang.constant.ConstantDescs.BSM_GET_STATIC_FINA	15	`MethodHandleDesc` representing `ConstantBootstraps.getStaticFinal`
java.lang.constant.ConstantDescs.CLASS_INIT_NAME	21	The special name of class initialization methods, "<clinit>".
java.lang.constant.ConstantDescs.FALSE	15	Nominal descriptor representing the constant Boolean.FALSE

java.lang.constant.ConstantDescs.INIT_NAME	21	The special name of instance initialization methods, "<init>".
java.lang.constant.ConstantDescs.MTD_void	21	Nominal descriptor representing the method descriptor ()V, taking no argument and returning void.
java.lang.constant.ConstantDescs.TRUE	15	Nominal descriptor representing the constant Boolean.TRUE
java.lang.Double.PRECISION	19	The number of bits in the significand of a double value.
java.lang.Float.PRECISION	19	The number of bits in the significand of a float value.
java.lang.invoke.MethodHandles.Lookup.ORIGINAL	16	A single-bit mask representing original access which may contribute to the result of lookupModes.
java.lang.invoke.StringConcatFactory.MAX_INDY_CONCAT	21	Maximum number of argument slots in String Concat call.
java.lang.Math.TAU	19	The double value that is closer than any other to *tau* (τ), the ratio of the circumference of a circle to its radius.
java.lang.StrictMath.TAU	19	The double value that is closer than any other to *tau* (τ), the ratio of the circumference of a circle to its radius.
java.math.BigDecimal.TWO	19	The value 2, with a scale of 0.
java.security.spec.MGF1ParameterSpec.SHA3_224	16	The MGF1ParameterSpec uses a "SHA3-224" message digest.
java.security.spec.MGF1ParameterSpec.SHA3_256	16	The MGF1ParameterSpec uses a "SHA3-256" message digest.
java.security.spec.MGF1ParameterSpec.SHA3_384	16	The MGF1ParameterSpec uses a "SHA3-384" message digest.
java.security.spec.MGF1ParameterSpec.SHA3_512	16	The MGF1ParameterSpec uses a "SHA3-512" message digest.
java.security.spec.NamedParameterSpec.ED25519	15	The Ed25519 parameters
java.security.spec.NamedParameterSpec.ED448	15	The Ed448 parameters
java.text.NumberFormat.Field.PREFIX	12	Constant identifying the prefix field.
java.text.NumberFormat.Field.SUFFIX	12	Constant identifying the suffix field.
java.time.chrono.JapaneseEra.REIWA	13	The singleton instance for the 'Reiwa' era (2019-05-01 -) which has the value 3.
javax.security.auth.kerberos.KerberosPrincipal.KRB_NT_I	13	Enterprise name (alias)
javax.xml.crypto.dsig.CanonicalizationMethod.INCLUSIVE	13	The Canonical XML 1.1 (without comments)⧉ canonicalization method algorithm URI.
javax.xml.crypto.dsig.CanonicalizationMethod.INCLUSIVE	13	The Canonical XML 1.1 with comments⧉ canonicalization method algorithm URI.
javax.xml.crypto.dsig.SignatureMethod.ED25519	21	The ED25519⧉ signature method algorithm URI.
javax.xml.crypto.dsig.SignatureMethod.ED448	21	The ED448⧉ signature method algorithm URI.
javax.xml.crypto.dsig.SignatureMethod.RSA_PSS	17	The RSASSA-PSS⧉ signature method algorithm URI.
jdk.incubator.vector.VectorOperators.BIT_COUNT	19	Produce bitCount(a)
jdk.incubator.vector.VectorOperators.COMPRESS_BITS	19	Produce compress(a,n).
jdk.incubator.vector.VectorOperators.EXPAND_BITS	19	Produce expand(a,n).
jdk.incubator.vector.VectorOperators.LEADING_ZEROS_C(	19	Produce numberOfLeadingZeros(a)
jdk.incubator.vector.VectorOperators.REVERSE	19	Produce reverse(a)
jdk.incubator.vector.VectorOperators.REVERSE_BYTES	19	Produce reverseBytes(a)
jdk.incubator.vector.VectorOperators.TRAILING_ZEROS_C	19	Produce numberOfTrailingZeros(a)
jdk.net.ExtendedSocketOptions.SO_INCOMING_NAPI_ID	15	Identifies the receive queue that the last incoming packet for the socket was received on.

jdk.net.ExtendedSocketOptions.SO_PEERCRED	16	Unix Domain peer credentials.

New Methods

Method ⇕	Added in ⇕	Description
com.sun.jdi.request.ThreadDeathRequest.addPlatformTh	21	Restricts the events generated by this request to only platform threads.
com.sun.jdi.request.ThreadStartRequest.addPlatformThi	21	Restricts the events generated by this request to only platform threads.
com.sun.jdi.ThreadReference.isVirtual()	21	Returns `true` if the thread is a virtual thread.
com.sun.management.HotSpotDiagnosticMXBean.dumpT (String, HotSpotDiagnosticMXBean.ThreadDumpFormat)	21	Generate a thread dump to the given file in the given format.
com.sun.management.OperatingSystemMXBean.getCpuI	14	Returns the "recent cpu usage" for the operating environment.
com.sun.management.OperatingSystemMXBean.getFree	14	Returns the amount of free memory in bytes.
com.sun.management.OperatingSystemMXBean.getTota	14	Returns the total amount of memory in bytes.
com.sun.management.ThreadMXBean.getCurrentThread	14	Returns an approximation of the total amount of memory, in bytes, allocated in heap memory for the current thread.
com.sun.management.ThreadMXBean.getTotalThreadAll	21	Returns an approximation of the total amount of memory, in bytes, allocated in heap memory by all threads since the Java virtual machine started.
com.sun.net.httpserver.Filter.adaptRequest(String, UnaryOperator<Request>)	18	Returns a pre-processing Filter that inspects and possibly adapts the request state.
com.sun.net.httpserver.Filter.afterHandler(String, Consumer<HttpExchange>)	17	Returns a post-processing `Filter` with the given description and operation.
com.sun.net.httpserver.Filter.beforeHandler(String, Consumer<HttpExchange>)	17	Returns a pre-processing `Filter` with the given description and operation.
com.sun.net.httpserver.Headers.of(String...)	18	Returns an immutable `Headers` with the given name value pairs as its set of headers.
com.sun.net.httpserver.Headers.of(Map<String, List<String>>)	18	Returns an immutable `Headers` from the given `headers` with the same header names and values.
com.sun.net.httpserver.HttpServer.create (InetSocketAddress, int, String, HttpHandler, Filter...)	18	Creates an `HttpServer` instance with an initial context.
com.sun.net.httpserver.HttpsServer.create (InetSocketAddress, int, String, HttpHandler, Filter...)	18	Creates an `HttpsServer` instance with an initial context.
com.sun.source.doctree.DocTreeVisitor.visitEscape (EscapeTree, P)	21	Visits an `EscapeTree` node.
com.sun.source.doctree.DocTreeVisitor.visitSnippet (SnippetTree, P)	18	Visits a `SnippetTree` node.
com.sun.source.doctree.DocTreeVisitor.visitSpec (SpecTree, P)	20	Visits a `SpecTree` node.
com.sun.source.doctree.DocTreeVisitor.visitSystemProp (SystemPropertyTree, P)	12	Visits a `SystemPropertyTree` node.
com.sun.source.doctree.ReturnTree.isInline()	16	Returns whether this instance is an inline tag.
com.sun.source.doctree.ValueTree.getFormat()	20	Returns the format string, or `null` if none was provided.
com.sun.source.tree.CaseTree.getBody()	12	For case with kind CaseTree.CaseKind.RULE, returns the statement or expression after the arrow.
com.sun.source.tree.CaseTree.getCaseKind()	12	Returns the kind of this case.
com.sun.source.tree.CaseTree.getExpressions()	12	Returns the labels for this case.
com.sun.source.tree.CaseTree.getGuard()	21	The guard for the case.
com.sun.source.tree.CaseTree.getLabels()	21	Returns the labels for this case.

com.sun.source.tree.ClassTree.getPermitsClause()	17	Returns the subclasses permitted by this type declaration.
com.sun.source.tree.CompilationUnitTree.getModule()	17	Returns the module tree associated with this compilation unit, or `null` if there is no module declaration.
com.sun.source.tree.InstanceOfTree.getPattern()	16	Returns the tested pattern, or null if this instanceof does not use a pattern.
com.sun.source.tree.TreeVisitor.visitAnyPattern (AnyPatternTree, P)[PREVIEW]	21	Visits a `AnyPatternTree` node.
com.sun.source.tree.TreeVisitor.visitBindingPattern (BindingPatternTree, P)	16	Visits a `BindingPatternTree` node.
com.sun.source.tree.TreeVisitor.visitConstantCaseLabel (ConstantCaseLabelTree, P)	21	Visits a `ConstantCaseLabelTree` node.
com.sun.source.tree.TreeVisitor.visitDeconstructionPatte (DeconstructionPatternTree, P)	21	Visits a `DeconstructionPatternTree` node.
com.sun.source.tree.TreeVisitor.visitDefaultCaseLabel (DefaultCaseLabelTree, P)	21	Visits a `DefaultCaseLabelTree` node.
com.sun.source.tree.TreeVisitor.visitPatternCaseLabel (PatternCaseLabelTree, P)	21	Visits a `PatternCaseLabelTree` node.
com.sun.source.tree.TreeVisitor.visitSwitchExpression (SwitchExpressionTree, P)	12	Visits a `SwitchExpressionTree` node.
com.sun.source.tree.TreeVisitor.visitYield(YieldTree, P)	13	Visits a `YieldTree` node.
com.sun.source.util.DocTreeFactory.newEscapeTree (char)	21	Creates a new `EscapeTree` object, to represent an escaped character.
com.sun.source.util.DocTreeFactory.newReturnTree (boolean, List<? extends DocTree>)	16	Creates a new `ReturnTree` object, to represent a @return tag or {@return} tag.
com.sun.source.util.DocTreeFactory.newSnippetTree (List<? extends DocTree>, TextTree)	18	Creates a new `SnippetTree` object, to represent a {@snippet } tag.
com.sun.source.util.DocTreeFactory.newSpecTree (TextTree, List<? extends DocTree>)	20	Creates a new `SpecTree` object, to represent an @spec tag.
com.sun.source.util.DocTreeFactory.newSystemProperty (Name)	12	Creates a new `SystemPropertyTree` object, to represent a {@systemProperty } tag.
com.sun.source.util.DocTreeFactory.newValueTree (TextTree, ReferenceTree)	20	Creates a new `ValueTree` object, to represent a {@value } tag.
com.sun.source.util.DocTrees.getType(DocTreePath)	15	Returns the language model type referred to by the leaf node of the given `DocTreePath`, or null if unknown.
com.sun.source.util.DocTreeScanner.visitEscape (EscapeTree, P)	21	Visits an `EscapeTree` node.
com.sun.source.util.DocTreeScanner.visitSnippet (SnippetTree, P)	18	Visits a `SnippetTree` node.
com.sun.source.util.DocTreeScanner.visitSpec (SpecTree, P)	20	Visits a `SpecTree` node.
com.sun.source.util.DocTreeScanner.visitSystemProperty (SystemPropertyTree, P)	12	Visits a `SystemPropertyTree` node.
com.sun.source.util.JavacTask.setParameterNameProvide (ParameterNameProvider)	13	Sets the specified `ParameterNameProvider`.
com.sun.source.util.SimpleDocTreeVisitor.visitEscape (EscapeTree, P)	21	Visits an `EscapeTree` node.
com.sun.source.util.SimpleDocTreeVisitor.visitSnippet (SnippetTree, P)	18	Visits a `SnippetTree` node.
com.sun.source.util.SimpleDocTreeVisitor.visitSpec (SpecTree, P)	20	Visits a `SpecTree` node.
com.sun.source.util.SimpleDocTreeVisitor.visitSystemPro (SystemPropertyTree, P)	12	Visits a `SystemPropertyTree` node.

com.sun.source.util.SimpleTreeVisitor.visitAnyPattern (AnyPatternTree, P)^{PREVIEW}	21	Visits a `AnyPatternTree` node.
com.sun.source.util.SimpleTreeVisitor.visitBindingPattern (BindingPatternTree, P)	14	Visits a `BindingPatternTree` node.
com.sun.source.util.SimpleTreeVisitor.visitConstantCaseLabel (ConstantCaseLabelTree, P)	21	Visits a `ConstantCaseLabelTree` node.
com.sun.source.util.SimpleTreeVisitor.visitDeconstructio (DeconstructionPatternTree, P)	21	Visits a `DeconstructionPatternTree` node.
com.sun.source.util.SimpleTreeVisitor.visitDefaultCaseLabel (DefaultCaseLabelTree, P)	21	Visits a `DefaultCaseLabelTree` node.
com.sun.source.util.SimpleTreeVisitor.visitPatternCaseLabel (PatternCaseLabelTree, P)	21	Visits a `PatternCaseLabelTree` node.
com.sun.source.util.TreeScanner.visitAnyPattern (AnyPatternTree, P)^{PREVIEW}	21	Visits a `AnyPatternTree` node.
com.sun.source.util.TreeScanner.visitBindingPattern (BindingPatternTree, P)	14	Visits a `BindingPatternTree` node.
com.sun.source.util.TreeScanner.visitConstantCaseLabel (ConstantCaseLabelTree, P)	21	Visits a `ConstantCaseLabelTree` node.
com.sun.source.util.TreeScanner.visitDeconstructionPatt (DeconstructionPatternTree, P)	21	Visits a `DeconstructionPatternTree` node.
com.sun.source.util.TreeScanner.visitDefaultCaseLabel (DefaultCaseLabelTree, P)	21	Visits a `DefaultCaseLabelTree` node.
com.sun.source.util.TreeScanner.visitPatternCaseLabel (PatternCaseLabelTree, P)	21	Visits a `PatternCaseLabelTree` node.
java.io.Console.charset()	17	Returns the `Charset` object used for the `Console`.
java.io.InputStream.skipNBytes(long)	12	Skips over and discards exactly n bytes of data from this input stream.
java.io.ObjectInputFilter.allowFilter(Predicate<Class<? >>, ObjectInputFilter.Status)	17	Returns a filter that returns `Status.ALLOWED` if the predicate on the class is `true`.
java.io.ObjectInputFilter.Config.getSerialFilterFactory()	17	Returns the JVM-wide deserialization filter factory.
java.io.ObjectInputFilter.Config.setSerialFilterFactory (BinaryOperator<ObjectInputFilter>)	17	Set the JVM-wide deserialization filter factory.
java.io.ObjectInputFilter.merge(ObjectInputFilter, ObjectInputFilter)	17	Returns a filter that merges the status of a filter and another filter.
java.io.ObjectInputFilter.rejectFilter(Predicate<Class<? >>, ObjectInputFilter.Status)	17	Returns a filter that returns `Status.REJECTED` if the predicate on the class is `true`.
java.io.ObjectInputFilter.rejectUndecidedClass (ObjectInputFilter)	17	Returns a filter that invokes a given filter and maps `UNDECIDED` to `REJECTED` for classes, with some special cases, and otherwise returns the status.
java.io.PrintStream.charset()	18	Returns the charset used in this `PrintStream` instance.
java.io.PrintStream.write(byte[])	14	Writes all bytes from the specified byte array to this stream.
java.io.PrintStream.writeBytes(byte[])	14	Writes all bytes from the specified byte array to this stream.
java.lang.Boolean.describeConstable()	15	Returns an `Optional` containing the nominal descriptor for this instance.
java.lang.Byte.describeConstable()	15	Returns an `Optional` containing the nominal descriptor for this instance.
java.lang.Character.describeConstable()	15	Returns an `Optional` containing the nominal descriptor for this instance.
java.lang.Character.isEmoji(int)	21	Determines if the specified character (Unicode code point) is an Emoji.

java.lang.Character.isEmojiComponent(int)	21	Determines if the specified character (Unicode code point) is an Emoji Component.
java.lang.Character.isEmojiModifier(int)	21	Determines if the specified character (Unicode code point) is an Emoji Modifier.
java.lang.Character.isEmojiModifierBase(int)	21	Determines if the specified character (Unicode code point) is an Emoji Modifier Base.
java.lang.Character.isEmojiPresentation(int)	21	Determines if the specified character (Unicode code point) has the Emoji Presentation property by default.
java.lang.Character.isExtendedPictographic(int)	21	Determines if the specified character (Unicode code point) is an Extended Pictographic.
java.lang.CharSequence.isEmpty()	15	Returns `true` if this character sequence is empty.
java.lang.Class.accessFlags()	20	Returns an unmodifiable set of the access flags for this class, possibly empty.
java.lang.Class.arrayType()	12	Returns a `Class` for an array type whose component type is described by this Class.
java.lang.Class.componentType()	12	Returns the component type of this `Class`, if it describes an array type, or `null` otherwise.
java.lang.Class.describeConstable()	12	Returns a nominal descriptor for this instance, if one can be constructed, or an empty `Optional` if one cannot be.
java.lang.Class.descriptorString()	12	Returns the descriptor string of the entity (class, interface, array class, primitive type, or `void`) represented by this `Class` object.
java.lang.Class.getPermittedSubclasses()	17	Returns an array containing `Class` objects representing the direct subinterfaces or subclasses permitted to extend or implement this class or interface if it is sealed.
java.lang.Class.getRecordComponents()	16	Returns an array of `RecordComponent` objects representing all the record components of this record class, or `null` if this class is not a record class.
java.lang.Class.isHidden()	15	Returns `true` if and only if the underlying class is a hidden class.
java.lang.Class.isRecord()	16	Returns `true` if and only if this class is a record class.
java.lang.Class.isSealed()	17	Returns `true` if and only if this `Class` object represents a sealed class or interface.
java.lang.Class.isUnnamedClass()[PREVIEW]	21	Returns `true` if and only if the underlying class is an unnamed class.
java.lang.constant.ClassDesc.ofInternalName(String)	20	Returns a ClassDesc for a class or interface type, given the name of the class or interface in internal form, such as `"java/lang/String"`.
java.lang.constant.MethodTypeDesc.of(ClassDesc)	21	Returns a MethodTypeDesc with the given return type and no parameter types.
java.lang.constant.MethodTypeDesc.of(ClassDesc, List<ClassDesc>)	21	Returns a MethodTypeDesc given the return type and a list of parameter types.
java.lang.Double.describeConstable()	12	Returns an `Optional` containing the nominal descriptor for this instance, which is the instance itself.
java.lang.Double.resolveConstantDesc (MethodHandles.Lookup)	12	Resolves this instance as a `ConstantDesc`, the result of which is the instance itself.
java.lang.Enum.describeConstable()	12	Returns an enum descriptor `EnumDesc` for this instance, if one can be constructed, or an empty `Optional` if one cannot be.
java.lang.Enum.EnumDesc.of(ClassDesc, String)	12	Returns a nominal descriptor for the specified `enum` class and name
java.lang.Float.describeConstable()	12	Returns an `Optional` containing the nominal descriptor for this instance, which is the instance itself.
java.lang.Float.float16ToFloat(short)	20	Returns the `float` value closest to the numerical value of the argument, a floating-point binary16 value encoded in a `short`.

API		Description
java.lang.Float.floatToFloat16(float)	20	Returns the floating-point binary16 value, encoded in a short, closest in value to the argument.
java.lang.Float.resolveConstantDesc (MethodHandles.Lookup)	12	Resolves this instance as a `ConstantDesc`, the result of which is the instance itself.
java.lang.Integer.compress(int, int)	19	Returns the value obtained by compressing the bits of the specified int value, i, in accordance with the specified bit mask.
java.lang.Integer.describeConstable()	12	Returns an `Optional` containing the nominal descriptor for this instance, which is the instance itself.
java.lang.Integer.expand(int, int)	19	Returns the value obtained by expanding the bits of the specified int value, i, in accordance with the specified bit mask.
java.lang.Integer.resolveConstantDesc (MethodHandles.Lookup)	12	Resolves this instance as a `ConstantDesc`, the result of which is the instance itself.
java.lang.invoke.ConstantBootstraps.explicitCast (MethodHandles.Lookup, String, Class<?>, Object)	15	Applies a conversion from a source type to a destination type.
java.lang.invoke.MethodHandle.describeConstable()	12	Return a nominal descriptor for this instance, if one can be constructed, or an empty `Optional` if one cannot be.
java.lang.invoke.MethodHandles.classData (MethodHandles.Lookup, String, Class<T>)	16	Returns the *class data* associated with the lookup class of the given caller lookup object, or null.
java.lang.invoke.MethodHandles.classDataAt (MethodHandles.Lookup, String, Class<T>, int)	16	Returns the element at the specified index in the class data, if the class data associated with the lookup class of the given caller lookup object is a List.
java.lang.invoke.MethodHandles.collectCoordinates (VarHandle, int, MethodHandle)[PREVIEW]	19	Adapts a target var handle by pre-processing a sub-sequence of its coordinate values with a filter (a method handle).
java.lang.invoke.MethodHandles.dropCoordinates (VarHandle, int, Class<?>...)[PREVIEW]	19	Returns a var handle which will discard some dummy coordinates before delegating to the target var handle.
java.lang.invoke.MethodHandles.dropReturn (MethodHandle)	16	Drop the return value of the target handle (if any).
java.lang.invoke.MethodHandles.filterCoordinates (VarHandle, int, MethodHandle...)[PREVIEW]	19	Adapts a target var handle by pre-processing incoming coordinate values using unary filter functions.
java.lang.invoke.MethodHandles.filterValue(VarHandle, MethodHandle, MethodHandle)[PREVIEW]	19	Adapts a target var handle by pre-processing incoming and outgoing values using a pair of filter functions.
java.lang.invoke.MethodHandles.insertCoordinates (VarHandle, int, Object...)[PREVIEW]	19	Provides a target var handle with one or more *bound coordinates* in advance of the var handle's invocation.
java.lang.invoke.MethodHandles.Lookup.defineHiddenCl (byte[], boolean, MethodHandles.Lookup.ClassOption...)	15	Creates a *hidden* class or interface from bytes, returning a `Lookup` on the newly created class or interface.
java.lang.invoke.MethodHandles.Lookup.defineHiddenCl (byte[], Object, boolean, MethodHandles.Lookup.ClassOption...)	16	Creates a *hidden* class or interface from bytes with associated class data, returning a `Lookup` on the newly created class or interface.
java.lang.invoke.MethodHandles.Lookup.ensureInitialize (Class<T>)	15	Ensures that targetClass has been initialized.
java.lang.invoke.MethodHandles.Lookup.hasFullPrivilege	14	Returns true if this lookup has *full privilege access*, i.e.
java.lang.invoke.MethodHandles.Lookup.previousLookup	14	Reports a lookup class in another module that this lookup object was previously teleported from, or null.
java.lang.invoke.MethodHandles.memorySegmentViewV (ValueLayout)[PREVIEW]	19	Creates a var handle object, which can be used to dereference a memory segment[PREVIEW] at a given byte offset, using the provided value layout.
java.lang.invoke.MethodHandles.permuteCoordinates (VarHandle, List<Class<?>>, int...)[PREVIEW]	19	Provides a var handle which adapts the coordinate values of the target var handle, by re-arranging them so that the new coordinates match the provided ones.
java.lang.invoke.MethodType.describeConstable()	12	Returns a nominal descriptor for this instance, if one can be constructed, or an empty `Optional` if one cannot

		be.
java.lang.invoke.MethodType.descriptorString()	12	Returns the descriptor string for this method type.
java.lang.invoke.StringConcatFactory.makeConcatWithTe (List<String>, List<Class<?>>)^{PREVIEW}	21	Simplified concatenation method to facilitate `StringTemplate`^{PREVIEW} concatenation.
java.lang.invoke.StringConcatFactory.makeConcatWithTe (List<String>, List<Class<?>>, int)^{PREVIEW}	21	This method breaks up large concatenations into separate `MethodHandles` based on the number of slots required per `MethodHandle`.
java.lang.invoke.StringConcatFactory.makeConcatWithTe (List<String>, List<MethodHandle>, int)^{PREVIEW}	21	This method creates a `MethodHandle` expecting one input, the receiver of the supplied getters.
java.lang.invoke.VarHandle.describeConstable()	12	Return a nominal descriptor for this instance, if one can be constructed, or an empty `Optional` if one cannot be.
java.lang.invoke.VarHandle.hasInvokeExactBehavior()	16	Returns `true` if this VarHandle has *invoke-exact behavior*.
java.lang.invoke.VarHandle.withInvokeBehavior()	16	Returns a VarHandle, with access to the same variable(s) as this VarHandle, but whose invocation behavior of access mode methods is adjusted to *invoke behavior*.
java.lang.invoke.VarHandle.withInvokeExactBehavior()	16	Returns a VarHandle, with access to the same variable(s) as this VarHandle, but whose invocation behavior of access mode methods is adjusted to *invoke-exact behavior*.
java.lang.Long.compress(long, long)	19	Returns the value obtained by compressing the bits of the specified `long` value, i, in accordance with the specified bit mask.
java.lang.Long.describeConstable()	12	Returns an `Optional` containing the nominal descriptor for this instance, which is the instance itself.
java.lang.Long.expand(long, long)	19	Returns the value obtained by expanding the bits of the specified `long` value, i, in accordance with the specified bit mask.
java.lang.Long.resolveConstantDesc (MethodHandles.Lookup)	12	Resolves this instance as a `ConstantDesc`, the result of which is the instance itself.
java.lang.Math.absExact(int)	15	Returns the mathematical absolute value of an `int` value if it is exactly representable as an `int`, throwing `ArithmeticException` if the result overflows the positive `int` range.
java.lang.Math.absExact(long)	15	Returns the mathematical absolute value of an `long` value if it is exactly representable as an `long`, throwing `ArithmeticException` if the result overflows the positive `long` range.
java.lang.Math.ceilDiv(int, int)	18	Returns the smallest (closest to negative infinity) `int` value that is greater than or equal to the algebraic quotient.
java.lang.Math.ceilDiv(long, int)	18	Returns the smallest (closest to negative infinity) `long` value that is greater than or equal to the algebraic quotient.
java.lang.Math.ceilDiv(long, long)	18	Returns the smallest (closest to negative infinity) `long` value that is greater than or equal to the algebraic quotient.
java.lang.Math.ceilDivExact(int, int)	18	Returns the smallest (closest to negative infinity) `int` value that is greater than or equal to the algebraic quotient.
java.lang.Math.ceilDivExact(long, long)	18	Returns the smallest (closest to negative infinity) `long` value that is greater than or equal to the algebraic quotient.
java.lang.Math.ceilMod(int, int)	18	Returns the ceiling modulus of the `int` arguments.
java.lang.Math.ceilMod(long, int)	18	Returns the ceiling modulus of the `long` and `int` arguments.
java.lang.Math.ceilMod(long, long)	18	Returns the ceiling modulus of the `long` arguments.

java.lang.Math.clamp(double, double, double)	21	Clamps the value to fit between min and max.
java.lang.Math.clamp(float, float, float)	21	Clamps the value to fit between min and max.
java.lang.Math.clamp(long, int, int)	21	Clamps the value to fit between min and max.
java.lang.Math.clamp(long, long, long)	21	Clamps the value to fit between min and max.
java.lang.Math.divideExact(int, int)	18	Returns the quotient of the arguments, throwing an exception if the result overflows an `int`.
java.lang.Math.divideExact(long, long)	18	Returns the quotient of the arguments, throwing an exception if the result overflows a `long`.
java.lang.Math.floorDivExact(int, int)	18	Returns the largest (closest to positive infinity) `int` value that is less than or equal to the algebraic quotient.
java.lang.Math.floorDivExact(long, long)	18	Returns the largest (closest to positive infinity) `long` value that is less than or equal to the algebraic quotient.
java.lang.Math.unsignedMultiplyHigh(long, long)	18	Returns as a `long` the most significant 64 bits of the unsigned 128-bit product of two unsigned 64-bit factors.
java.lang.Module.isNativeAccessEnabled()[PREVIEW]	20	Returns `true` if this module can access *restricted* methods.
java.lang.module.ModuleDescriptor.accessFlags()	20	Returns the set of the module flags.
java.lang.module.ModuleDescriptor.Exports.accessFlags(	20	Returns the set of the module export flags for this module descriptor.
java.lang.module.ModuleDescriptor.Opens.accessFlags()	20	Returns the set of the module opens flags.
java.lang.module.ModuleDescriptor.Requires.accessFlags	20	Returns the set of the module requires flags.
java.lang.ModuleLayer.Controller.enableNativeAccess (Module)[PREVIEW]	20	Enables native access for a module in the layer if the caller's module has native access.
java.lang.Process.errorReader()	17	Returns a `BufferedReader` connected to the standard error of the process.
java.lang.Process.errorReader(Charset)	17	Returns a `BufferedReader` connected to the standard error of this process using a Charset.
java.lang.Process.inputReader()	17	Returns a `BufferedReader` connected to the standard output of the process.
java.lang.Process.inputReader(Charset)	17	Returns a `BufferedReader` connected to the standard output of this process using a Charset.
java.lang.Process.outputWriter()	17	Returns a `BufferedWriter` connected to the normal input of the process using the native encoding.
java.lang.Process.outputWriter(Charset)	17	Returns a `BufferedWriter` connected to the normal input of the process using a Charset.
java.lang.ref.Reference.refersTo(T)	16	Tests if the referent of this reference object is `obj`.
java.lang.reflect.Executable.accessFlags()	20	Returns an unmodifiable set of the access flags for the executable represented by this object, possibly empty.
java.lang.reflect.Field.accessFlags()	20	Returns an unmodifiable set of the access flags for this field, possibly empty.
java.lang.reflect.InvocationHandler.invokeDefault (Object, Method, Object...)	16	Invokes the specified default method on the given `proxy` instance with the given parameters.
java.lang.reflect.Member.accessFlags()	20	Returns an unmodifiable set of the access flags for this member, possibly empty.
java.lang.reflect.Parameter.accessFlags()	20	Returns an unmodifiable set of the access flags for the parameter represented by this object, possibly empty.
java.lang.Short.describeConstable()	15	Returns an `Optional` containing the nominal descriptor for this instance.
java.lang.StrictMath.absExact(int)	15	Returns the mathematical absolute value of an `int` value if it is exactly representable as an `int`, throwing `ArithmeticException` if the result overflows the positive `int` range.

java.lang.StrictMath.absExact(long)	15	Returns the mathematical absolute value of an `long` value if it is exactly representable as an `long`, throwing `ArithmeticException` if the result overflows the positive `long` range.
java.lang.StrictMath.ceilDiv(int, int)	18	Returns the smallest (closest to negative infinity) `int` value that is greater than or equal to the algebraic quotient.
java.lang.StrictMath.ceilDiv(long, int)	18	Returns the smallest (closest to negative infinity) `long` value that is greater than or equal to the algebraic quotient.
java.lang.StrictMath.ceilDiv(long, long)	18	Returns the smallest (closest to negative infinity) `long` value that is greater than or equal to the algebraic quotient.
java.lang.StrictMath.ceilDivExact(int, int)	18	Returns the smallest (closest to negative infinity) `int` value that is greater than or equal to the algebraic quotient.
java.lang.StrictMath.ceilDivExact(long, long)	18	Returns the smallest (closest to negative infinity) `long` value that is greater than or equal to the algebraic quotient.
java.lang.StrictMath.ceilMod(int, int)	18	Returns the ceiling modulus of the `int` arguments.
java.lang.StrictMath.ceilMod(long, int)	18	Returns the ceiling modulus of the `long` and `int` arguments.
java.lang.StrictMath.ceilMod(long, long)	18	Returns the ceiling modulus of the `long` arguments.
java.lang.StrictMath.clamp(double, double, double)	21	Clamps the value to fit between min and max.
java.lang.StrictMath.clamp(float, float, float)	21	Clamps the value to fit between min and max.
java.lang.StrictMath.clamp(long, int, int)	21	Clamps the value to fit between min and max.
java.lang.StrictMath.clamp(long, long, long)	21	Clamps the value to fit between min and max.
java.lang.StrictMath.decrementExact(int)	14	Returns the argument decremented by one, throwing an exception if the result overflows an `int`.
java.lang.StrictMath.decrementExact(long)	14	Returns the argument decremented by one, throwing an exception if the result overflows a `long`.
java.lang.StrictMath.divideExact(int, int)	18	Returns the quotient of the arguments, throwing an exception if the result overflows an `int`.
java.lang.StrictMath.divideExact(long, long)	18	Returns the quotient of the arguments, throwing an exception if the result overflows a `long`.
java.lang.StrictMath.floorDivExact(int, int)	18	Returns the largest (closest to positive infinity) `int` value that is less than or equal to the algebraic quotient.
java.lang.StrictMath.floorDivExact(long, long)	18	Returns the largest (closest to positive infinity) `long` value that is less than or equal to the algebraic quotient.
java.lang.StrictMath.incrementExact(int)	14	Returns the argument incremented by one, throwing an exception if the result overflows an `int`.
java.lang.StrictMath.incrementExact(long)	14	Returns the argument incremented by one, throwing an exception if the result overflows a `long`.
java.lang.StrictMath.negateExact(int)	14	Returns the negation of the argument, throwing an exception if the result overflows an `int`.
java.lang.StrictMath.negateExact(long)	14	Returns the negation of the argument, throwing an exception if the result overflows a `long`.
java.lang.StrictMath.unsignedMultiplyHigh(long, long)	18	Returns as a `long` the most significant 64 bits of the unsigned 128-bit product of two unsigned 64-bit factors.
java.lang.String.describeConstable()	12	Returns an `Optional` containing the nominal descriptor for this instance, which is the instance itself.
java.lang.String.formatted(Object...)	15	Formats using this string as the format string, and the supplied arguments.
java.lang.String.indent(int)	12	Adjusts the indentation of each line of this string based on the value of `n`, and normalizes line termination characters.

java.lang.String.indexOf(int, int, int)	21	Returns the index within this string of the first occurrence of the specified character, starting the search at `beginIndex` and stopping before `endIndex`.
java.lang.String.indexOf(String, int, int)	21	Returns the index of the first occurrence of the specified substring within the specified index range of `this` string.
java.lang.String.resolveConstantDesc (MethodHandles.Lookup)	12	Resolves this instance as a `ConstantDesc`, the result of which is the instance itself.
java.lang.String.splitWithDelimiters(String, int)	21	Splits this string around matches of the given regular expression and returns both the strings and the matching delimiters.
java.lang.String.stripIndent()	15	Returns a string whose value is this string, with incidental white space removed from the beginning and end of every line.
java.lang.String.transform(Function<? super String, ? extends R>)	12	This method allows the application of a function to `this` string.
java.lang.String.translateEscapes()	15	Returns a string whose value is this string, with escape sequences translated as if in a string literal.
java.lang.StringBuffer.repeat(int, int)	21	
java.lang.StringBuffer.repeat(CharSequence, int)	21	
java.lang.StringBuilder.repeat(int, int)	21	
java.lang.StringBuilder.repeat(CharSequence, int)	21	
java.lang.Thread.isVirtual()	21	Returns `true` if this thread is a virtual thread.
java.lang.Thread.join(Duration)	19	Waits for this thread to terminate for up to the given waiting duration.
java.lang.Thread.ofPlatform()	21	Returns a builder for creating a platform `Thread` or `ThreadFactory` that creates platform threads.
java.lang.Thread.ofVirtual()	21	Returns a builder for creating a virtual `Thread` or `ThreadFactory` that creates virtual threads.
java.lang.Thread.sleep(Duration)	19	Causes the currently executing thread to sleep (temporarily cease execution) for the specified duration, subject to the precision and accuracy of system timers and schedulers.
java.lang.Thread.startVirtualThread(Runnable)	21	Creates a virtual thread to execute a task and schedules it to execute.
java.lang.Thread.threadId()	19	Returns the identifier of this Thread.
java.math.BigInteger.parallelMultiply(BigInteger)	19	Returns a BigInteger whose value is (`this * val`).
java.net.DatagramSocket.joinGroup(SocketAddress, NetworkInterface)	17	Joins a multicast group.
java.net.DatagramSocket.leaveGroup(SocketAddress, NetworkInterface)	17	Leave a multicast group on a specified local interface.
java.net.http.HttpClient.awaitTermination(Duration)	21	Blocks until all operations have completed execution after a shutdown request, or the `duration` elapses, or the current thread is interrupted, whichever happens first.
java.net.http.HttpClient.Builder.localAddress (InetAddress)	19	Binds the socket to this local address when creating connections for sending requests.
java.net.http.HttpClient.close()	21	Initiates an orderly shutdown in which requests previously submitted to `send` or `sendAsync` are run to completion, but no new request will be accepted.
java.net.http.HttpClient.isTerminated()	21	Returns `true` if all operations have completed following a shutdown.
java.net.http.HttpClient.shutdown()	21	Initiates an orderly shutdown in which requests previously submitted with `send` or `sendAsync` are run to completion, but no new request will be accepted.
java.net.http.HttpClient.shutdownNow()	21	This method attempts to initiate an immediate shutdown.
java.net.http.HttpRequest.BodyPublishers.concat (HttpRequest.BodyPublisher...)	16	Returns a `BodyPublisher` that publishes a request body consisting of the concatenation of the request bodies

		published by a sequence of publishers.
java.net.http.HttpRequest.Builder.HEAD()	18	Sets the request method of this builder to HEAD.
java.net.http.HttpRequest.newBuilder(HttpRequest, BiPredicate<String, String>)	16	Creates a `Builder` whose initial state is copied from an existing `HttpRequest`.
java.net.SecureCacheResponse.getSSLSession()	12	Returns an `Optional` containing the `SSLSession` in use on the original connection that retrieved the network resource.
java.net.URL.of(URI, URLStreamHandler)	20	Creates a URL from a URI, as if by invoking `uri.toURL()`, but associating it with the given `URLStreamHandler`, if allowed.
java.nio.Buffer.slice(int, int)	13	Creates a new buffer whose content is a shared subsequence of this buffer's content.
java.nio.ByteBuffer.get(int, byte[])	13	Absolute bulk *get* method.
java.nio.ByteBuffer.get(int, byte[], int, int)	13	Absolute bulk *get* method.
java.nio.ByteBuffer.put(int, byte[])	13	Absolute bulk *put* method *(optional operation)*.
java.nio.ByteBuffer.put(int, byte[], int, int)	13	Absolute bulk *put* method *(optional operation)*.
java.nio.ByteBuffer.put(int, ByteBuffer, int, int)	16	Absolute bulk *put* method *(optional operation)*.
java.nio.ByteBuffer.slice(int, int)	13	Creates a new byte buffer whose content is a shared subsequence of this buffer's content.
java.nio.channels.FileChannel.map (FileChannel.MapMode, long, long, Arena)^{PREVIEW}	19	Maps a region of this channel's file into a new mapped memory segment, with the given offset, size and arena.
java.nio.channels.ServerSocketChannel.open (ProtocolFamily)	15	Opens a server-socket channel.
java.nio.channels.SocketChannel.open(ProtocolFamily)	15	Opens a socket channel.
java.nio.channels.spi.SelectorProvider.openServerSocket (ProtocolFamily)	15	Opens a server-socket channel.
java.nio.channels.spi.SelectorProvider.openSocketChann (ProtocolFamily)	15	Opens a socket channel.
java.nio.CharBuffer.get(int, char[])	13	Absolute bulk *get* method.
java.nio.CharBuffer.get(int, char[], int, int)	13	Absolute bulk *get* method.
java.nio.CharBuffer.isEmpty()	15	Returns `true` if this character buffer is empty.
java.nio.CharBuffer.put(int, char[])	13	Absolute bulk *put* method *(optional operation)*.
java.nio.CharBuffer.put(int, char[], int, int)	13	Absolute bulk *put* method *(optional operation)*.
java.nio.CharBuffer.put(int, CharBuffer, int, int)	16	Absolute bulk *put* method *(optional operation)*.
java.nio.CharBuffer.slice(int, int)	13	Creates a new char buffer whose content is a shared subsequence of this buffer's content.
java.nio.charset.Charset.forName(String, Charset)	18	Returns a charset object for the named charset.
java.nio.DoubleBuffer.get(int, double[])	13	Absolute bulk *get* method.
java.nio.DoubleBuffer.get(int, double[], int, int)	13	Absolute bulk *get* method.
java.nio.DoubleBuffer.put(int, double[])	13	Absolute bulk *put* method *(optional operation)*.
java.nio.DoubleBuffer.put(int, double[], int, int)	13	Absolute bulk *put* method *(optional operation)*.
java.nio.DoubleBuffer.put(int, DoubleBuffer, int, int)	16	Absolute bulk *put* method *(optional operation)*.
java.nio.DoubleBuffer.slice(int, int)	13	Creates a new double buffer whose content is a shared subsequence of this buffer's content.
java.nio.file.Files.mismatch(Path, Path)	12	Finds and returns the position of the first mismatched byte in the content of two files, or `-1L` if there is no mismatch.
java.nio.file.FileSystems.newFileSystem(Path)	13	Constructs a new `FileSystem` to access the contents of a file as a file system.

java.nio.file.FileSystems.newFileSystem(Path, Map<String, ?>)	13	Constructs a new `FileSystem` to access the contents of a file as a file system.
java.nio.file.FileSystems.newFileSystem(Path, Map<String, ?>, ClassLoader)	13	Constructs a new `FileSystem` to access the contents of a file as a file system.
java.nio.file.spi.FileSystemProvider.exists(Path, LinkOption...)	20	Tests whether a file exists.
java.nio.file.spi.FileSystemProvider.readAttributesIfExist (Path, Class<A>, LinkOption...)	20	Reads a file's attributes as a bulk operation if it exists.
java.nio.FloatBuffer.get(int, float[])	13	Absolute bulk *get* method.
java.nio.FloatBuffer.get(int, float[], int, int)	13	Absolute bulk *get* method.
java.nio.FloatBuffer.put(int, float[])	13	Absolute bulk *put* method *(optional operation)*.
java.nio.FloatBuffer.put(int, float[], int, int)	13	Absolute bulk *put* method *(optional operation)*.
java.nio.FloatBuffer.put(int, FloatBuffer, int, int)	16	Absolute bulk *put* method *(optional operation)*.
java.nio.FloatBuffer.slice(int, int)	13	Creates a new float buffer whose content is a shared subsequence of this buffer's content.
java.nio.IntBuffer.get(int, int[])	13	Absolute bulk *get* method.
java.nio.IntBuffer.get(int, int[], int, int)	13	Absolute bulk *get* method.
java.nio.IntBuffer.put(int, int[])	13	Absolute bulk *put* method *(optional operation)*.
java.nio.IntBuffer.put(int, int[], int, int)	13	Absolute bulk *put* method *(optional operation)*.
java.nio.IntBuffer.put(int, IntBuffer, int, int)	16	Absolute bulk *put* method *(optional operation)*.
java.nio.IntBuffer.slice(int, int)	13	Creates a new int buffer whose content is a shared subsequence of this buffer's content.
java.nio.LongBuffer.get(int, long[])	13	Absolute bulk *get* method.
java.nio.LongBuffer.get(int, long[], int, int)	13	Absolute bulk *get* method.
java.nio.LongBuffer.put(int, long[])	13	Absolute bulk *put* method *(optional operation)*.
java.nio.LongBuffer.put(int, long[], int, int)	13	Absolute bulk *put* method *(optional operation)*.
java.nio.LongBuffer.put(int, LongBuffer, int, int)	16	Absolute bulk *put* method *(optional operation)*.
java.nio.LongBuffer.slice(int, int)	13	Creates a new long buffer whose content is a shared subsequence of this buffer's content.
java.nio.MappedByteBuffer.force(int, int)	13	Forces any changes made to a region of this buffer's content to be written to the storage device containing the mapped file.
java.nio.ShortBuffer.get(int, short[])	13	Absolute bulk *get* method.
java.nio.ShortBuffer.get(int, short[], int, int)	13	Absolute bulk *get* method.
java.nio.ShortBuffer.put(int, short[])	13	Absolute bulk *put* method *(optional operation)*.
java.nio.ShortBuffer.put(int, short[], int, int)	13	Absolute bulk *put* method *(optional operation)*.
java.nio.ShortBuffer.put(int, ShortBuffer, int, int)	16	Absolute bulk *put* method *(optional operation)*.
java.nio.ShortBuffer.slice(int, int)	13	Creates a new short buffer whose content is a shared subsequence of this buffer's content.
java.security.KeyStore.getAttributes(String)	18	Retrieves the attributes associated with the given alias.
java.security.KeyStoreSpi.engineGetAttributes(String)	18	Retrieves the attributes associated with the given alias.
java.text.DecimalFormatSymbols.getLocale()	19	Returns locale used to create this instance.
java.text.DecimalFormatSymbols.getMonetaryGroupingS	15	Gets the character used for grouping separator for currencies.
java.text.DecimalFormatSymbols.setMonetaryGroupingS (char)	15	Sets the character used for grouping separator for currencies.
java.text.NumberFormat.getCompactNumberInstance()	12	Returns a compact number format for the default `FORMAT` locale with `"SHORT"` format style.

java.text.NumberFormat.getCompactNumberInstance (Locale, NumberFormat.Style)	12	Returns a compact number format for the specified `locale` and `formatStyle`.
java.text.spi.NumberFormatProvider.getCompactNumber (Locale, NumberFormat.Style)	12	Returns a new `NumberFormat` instance which formats a number in its compact form for the specified `locale` and `formatStyle`.
java.time.chrono.Chronology.isIsoBased()	19	Checks if this chronology is ISO based.
java.time.chrono.IsoChronology.isIsoBased()	19	`IsoChronology` is an ISO based chronology, which supports fields in `IsoFields`, such as `DAY_OF_QUARTER` and `QUARTER_OF_YEAR`.
java.time.chrono.JapaneseChronology.isIsoBased()	19	`JapaneseChronology` is an ISO based chronology, which supports fields in `IsoFields`, such as `DAY_OF_QUARTER` and `QUARTER_OF_YEAR`.
java.time.chrono.MinguoChronology.isIsoBased()	19	`MinguoChronology` is an ISO based chronology, which supports fields in `IsoFields`, such as `DAY_OF_QUARTER` and `QUARTER_OF_YEAR`.
java.time.chrono.ThaiBuddhistChronology.isIsoBased()	19	`ThaiBuddhistChronology` is an ISO based chronology, which supports fields in `IsoFields`, such as `DAY_OF_QUARTER` and `QUARTER_OF_YEAR`.
java.time.Duration.isPositive()	18	Checks if this duration is positive, excluding zero.
java.time.format.DateTimeFormatter.ofLocalizedPattern (String)	19	Creates a locale specific formatter derived from the requested template for the ISO chronology.
java.time.format.DateTimeFormatterBuilder.appendDayF (TextStyle)	16	Appends the day period text to the formatter.
java.time.format.DateTimeFormatterBuilder.appendLoca (String)	19	Appends a localized pattern to the formatter using the requested template.
java.time.format.DateTimeFormatterBuilder.getLocalizec (String, Chronology, Locale)	19	Returns the formatting pattern for the requested template for a locale and chronology.
java.util.ArrayList.addFirst(E)	21	Adds an element as the first element of this collection (optional operation).
java.util.ArrayList.addLast(E)	21	Adds an element as the last element of this collection (optional operation).
java.util.ArrayList.getFirst()	21	Gets the first element of this collection.
java.util.ArrayList.getLast()	21	Gets the last element of this collection.
java.util.ArrayList.removeFirst()	21	Removes and returns the first element of this collection (optional operation).
java.util.ArrayList.removeLast()	21	Removes and returns the last element of this collection (optional operation).
java.util.Collections.newSequencedSetFromMap (SequencedMap<E, Boolean>)	21	Returns a sequenced set backed by the specified map.
java.util.Collections.shuffle(List<?>, RandomGenerator)	21	Randomly permute the specified list using the specified source of randomness.
java.util.Collections.unmodifiableSequencedCollection (SequencedCollection<? extends T>)	21	Returns an unmodifiable view of the specified `SequencedCollection`.
java.util.Collections.unmodifiableSequencedMap (SequencedMap<? extends K, ? extends V>)	21	Returns an unmodifiable view of the specified `SequencedMap`.
java.util.Collections.unmodifiableSequencedSet (SequencedSet<? extends T>)	21	Returns an unmodifiable view of the specified `SequencedSet`.
java.util.concurrent.CompletionStage.exceptionallyAsyn (Function<Throwable, ? extends T>)	12	Returns a new CompletionStage that, when this stage completes exceptionally, is executed with this stage's exception as the argument to the supplied function, using this stage's default asynchronous execution facility.
java.util.concurrent.CompletionStage.exceptionallyAsyn (Function<Throwable, ? extends T>, Executor)	12	Returns a new CompletionStage that, when this stage completes exceptionally, is executed with this stage's

		exception as the argument to the supplied function, using the supplied Executor.
java.util.concurrent.CompletionStage.exceptionallyComp (Function<Throwable, ? extends CompletionStage<T>>)	12	Returns a new CompletionStage that, when this stage completes exceptionally, is composed using the results of the supplied function applied to this stage's exception.
java.util.concurrent.CompletionStage.exceptionallyComp (Function<Throwable, ? extends CompletionStage<T>>)	12	Returns a new CompletionStage that, when this stage completes exceptionally, is composed using the results of the supplied function applied to this stage's exception, using this stage's default asynchronous execution facility.
java.util.concurrent.CompletionStage.exceptionallyComp (Function<Throwable, ? extends CompletionStage<T>>, Executor)	12	Returns a new CompletionStage that, when this stage completes exceptionally, is composed using the results of the supplied function applied to this stage's exception, using the supplied Executor.
java.util.concurrent.ConcurrentSkipListMap.putFirst(K, V)	21	Throws UnsupportedOperationException.
java.util.concurrent.ConcurrentSkipListMap.putLast(K, V)	21	Throws UnsupportedOperationException.
java.util.concurrent.ConcurrentSkipListSet.addFirst(E)	21	Throws UnsupportedOperationException.
java.util.concurrent.ConcurrentSkipListSet.addLast(E)	21	Throws UnsupportedOperationException.
java.util.concurrent.CopyOnWriteArrayList.addFirst(E)	21	Adds an element as the first element of this collection (optional operation).
java.util.concurrent.CopyOnWriteArrayList.addLast(E)	21	Adds an element as the last element of this collection (optional operation).
java.util.concurrent.CopyOnWriteArrayList.getFirst()	21	Gets the first element of this collection.
java.util.concurrent.CopyOnWriteArrayList.getLast()	21	Gets the last element of this collection.
java.util.concurrent.CopyOnWriteArrayList.removeFirst()	21	Removes and returns the first element of this collection (optional operation).
java.util.concurrent.CopyOnWriteArrayList.removeLast()	21	Removes and returns the last element of this collection (optional operation).
java.util.concurrent.CopyOnWriteArrayList.reversed()	21	Returns a reverse-ordered view of this collection.
java.util.concurrent.Executors.newThreadPerTaskExecut (ThreadFactory)	21	Creates an Executor that starts a new Thread for each task.
java.util.concurrent.Executors.newVirtualThreadPerTask	21	Creates an Executor that starts a new virtual Thread for each task.
java.util.concurrent.ExecutorService.close()	19	Initiates an orderly shutdown in which previously submitted tasks are executed, but no new tasks will be accepted.
java.util.concurrent.ForkJoinPool.close()	19	Unless this is the ForkJoinPool.commonPool(), initiates an orderly shutdown in which previously submitted tasks are executed, but no new tasks will be accepted, and waits until all tasks have completed execution and the executor has terminated.
java.util.concurrent.ForkJoinPool.externalSubmit (ForkJoinTask<T>)	20	Submits the given task as if submitted from a non-ForkJoinTask client.
java.util.concurrent.ForkJoinPool.lazySubmit (ForkJoinTask<T>)	19	Submits the given task without guaranteeing that it will eventually execute in the absence of available active threads.
java.util.concurrent.ForkJoinPool.setParallelism(int)	19	Changes the target parallelism of this pool, controlling the future creation, use, and termination of worker threads.
java.util.concurrent.ForkJoinTask.adaptInterruptible (Callable<? extends T>)	19	Returns a new ForkJoinTask that performs the call method of the given Callable as its action, and returns its result upon ForkJoinTask.join(), translating any checked exceptions encountered into RuntimeException.
java.util.concurrent.ForkJoinTask.quietlyJoin(long, TimeUnit)	19	Tries to join this task, returning true if it completed (possibly exceptionally) before the given timeout and

		the current thread has not been interrupted.	
java.util.concurrent.ForkJoinTask.quietlyJoinUninterrupti (long, TimeUnit)	19	Tries to join this task, returning true if it completed (possibly exceptionally) before the given timeout.	
java.util.concurrent.ForkJoinWorkerThread.getQueuedTa	20	Returns a (non-negative) estimate of the number of tasks in the thread's queue.	
java.util.concurrent.Future.exceptionNow()	19	Returns the exception thrown by the task, without waiting.	
java.util.concurrent.Future.resultNow()	19	Returns the computed result, without waiting.	
java.util.concurrent.Future.state()	19	Returns the computation state.	
java.util.concurrent.locks.LockSupport.setCurrentBlocke (Object)	14	Sets the object to be returned by invocations of `getBlocker` for the current thread.	
java.util.Deque.reversed()	21	Returns a reverse-ordered view of this collection.	
java.util.HashMap.newHashMap(int)	19	Creates a new, empty HashMap suitable for the expected number of mappings.	
java.util.HashSet.newHashSet(int)	19	Creates a new, empty HashSet suitable for the expected number of elements.	
java.util.LinkedHashMap.newLinkedHashMap(int)	19	Creates a new, empty, insertion-ordered LinkedHashMap suitable for the expected number of mappings.	
java.util.LinkedHashMap.putFirst(K, V)	21	Inserts the given mapping into the map if it is not already present, or replaces the value of a mapping if it is already present (optional operation).	
java.util.LinkedHashMap.putLast(K, V)	21	Inserts the given mapping into the map if it is not already present, or replaces the value of a mapping if it is already present (optional operation).	
java.util.LinkedHashMap.reversed()	21	Returns a reverse-ordered view of this map.	
java.util.LinkedHashMap.sequencedEntrySet()	21	Returns a `SequencedSet` view of this map's `entrySet`.	
java.util.LinkedHashMap.sequencedKeySet()	21	Returns a `SequencedSet` view of this map's `keySet`.	
java.util.LinkedHashMap.sequencedValues()	21	Returns a `SequencedCollection` view of this map's `values` collection.	
java.util.LinkedHashSet.addFirst(E)	21	Adds an element as the first element of this collection (optional operation).	
java.util.LinkedHashSet.addLast(E)	21	Adds an element as the last element of this collection (optional operation).	
java.util.LinkedHashSet.getFirst()	21	Gets the first element of this collection.	
java.util.LinkedHashSet.getLast()	21	Gets the last element of this collection.	
java.util.LinkedHashSet.newLinkedHashSet(int)	19	Creates a new, empty LinkedHashSet suitable for the expected number of elements.	
java.util.LinkedHashSet.removeFirst()	21	Removes and returns the first element of this collection (optional operation).	
java.util.LinkedHashSet.removeLast()	21	Removes and returns the last element of this collection (optional operation).	
java.util.LinkedHashSet.reversed()	21	Returns a reverse-ordered view of this collection.	
java.util.LinkedList.reversed()	21	Returns a reverse-ordered view of this collection.	
java.util.List.addFirst(E)	21	Adds an element as the first element of this collection (optional operation).	
java.util.List.addLast(E)	21	Adds an element as the last element of this collection (optional operation).	
java.util.List.getFirst()	21	Gets the first element of this collection.	
java.util.List.getLast()	21	Gets the last element of this collection.	
java.util.List.removeFirst()	21	Removes and returns the first element of this collection (optional operation).	

java.util.List.removeLast()	21	Removes and returns the last element of this collection (optional operation).
java.util.List.reversed()	21	Returns a reverse-ordered view of this collection.
java.util.Locale.availableLocales()	21	Returns a stream of installed locales.
java.util.Locale.caseFoldLanguageTag(String)	21	Returns a case folded IETF BCP 47 language tag.
java.util.Locale.of(String)	19	Obtains a locale from a language code.
java.util.Locale.of(String, String)	19	Obtains a locale from language and country.
java.util.Locale.of(String, String, String)	19	Obtains a locale from language, country and variant.
java.util.logging.LogRecord.getLongThreadID()	16	Get a thread identifier for the thread where message originated
java.util.logging.LogRecord.setLongThreadID(long)	16	Set an identifier for the thread where the message originated.
java.util.Map.Entry.copyOf(Map.Entry<? extends K, ? extends V>)	17	Returns a copy of the given `Map.Entry`.
java.util.NavigableMap.reversed()	21	Returns a reverse-ordered view of this map.
java.util.NavigableSet.removeFirst()	21	Removes and returns the first element of this collection (optional operation).
java.util.NavigableSet.removeLast()	21	Removes and returns the last element of this collection (optional operation).
java.util.NavigableSet.reversed()	21	Returns a reverse-ordered view of this collection.
java.util.Objects.checkFromIndexSize(long, long, long)	16	Checks if the sub-range from `fromIndex` (inclusive) to `fromIndex + size` (exclusive) is within the bounds of range from 0 (inclusive) to `length` (exclusive).
java.util.Objects.checkFromToIndex(long, long, long)	16	Checks if the sub-range from `fromIndex` (inclusive) to `toIndex` (exclusive) is within the bounds of range from 0 (inclusive) to `length` (exclusive).
java.util.Objects.checkIndex(long, long)	16	Checks if the `index` is within the bounds of the range from 0 (inclusive) to `length` (exclusive).
java.util.Objects.toIdentityString(Object)	19	Returns a string equivalent to the string returned by `Object.toString` if that method and `hashCode` are not overridden.
java.util.Random.from(RandomGenerator)	19	Returns an instance of `Random` that delegates method calls to the `RandomGenerator` argument.
java.util.regex.Matcher.hasMatch()	20	Returns whether `this` contains a valid match from a previous match or find operation.
java.util.regex.Matcher.namedGroups()	20	Returns an unmodifiable map from capturing group names to group numbers.
java.util.regex.MatchResult.end(String)	20	Returns the offset after the last character of the subsequence captured by the given named-capturing group during the previous match operation.
java.util.regex.MatchResult.group(String)	20	Returns the input subsequence captured by the given named-capturing group during the previous match operation.
java.util.regex.MatchResult.hasMatch()	20	Returns whether `this` contains a valid match from a previous match or find operation.
java.util.regex.MatchResult.namedGroups()	20	Returns an unmodifiable map from capturing group names to group numbers.
java.util.regex.MatchResult.start(String)	20	Returns the start index of the subsequence captured by the given named-capturing group during the previous match operation.
java.util.regex.Pattern.namedGroups()	20	Returns an unmodifiable map from capturing group names to group numbers.
java.util.regex.Pattern.splitWithDelimiters (CharSequence, int)	21	Splits the given input sequence around matches of this pattern and returns both the strings and the matching delimiters.

java.util.SortedMap.putFirst(K, V)	21	Throws `UnsupportedOperationException`.
java.util.SortedMap.putLast(K, V)	21	Throws `UnsupportedOperationException`.
java.util.SortedMap.reversed()	21	Returns a reverse-ordered view of this map.
java.util.SortedSet.addFirst(E)	21	Throws `UnsupportedOperationException`.
java.util.SortedSet.addLast(E)	21	Throws `UnsupportedOperationException`.
java.util.SortedSet.getFirst()	21	Gets the first element of this collection.
java.util.SortedSet.getLast()	21	Gets the last element of this collection.
java.util.SortedSet.removeFirst()	21	Removes and returns the first element of this collection (optional operation).
java.util.SortedSet.removeLast()	21	Removes and returns the last element of this collection (optional operation).
java.util.SortedSet.reversed()	21	Returns a reverse-ordered view of this collection.
java.util.spi.ToolProvider.description()	19	Returns a short description of the tool, or an empty `Optional` if no description is available.
java.util.SplittableRandom.split (RandomGenerator.SplittableGenerator)	17	Returns a new pseudorandom number generator, split off from this one, that implements the `RandomGenerator` and `RandomGenerator.SplittableGenerator` interfaces.
java.util.SplittableRandom.splits()	17	Returns an effectively unlimited stream of new pseudorandom number generators, each of which implements the `RandomGenerator.SplittableGenerator` interface.
java.util.SplittableRandom.splits(long)	17	Returns a stream producing the given `streamSize` number of new pseudorandom number generators, each of which implements the `RandomGenerator.SplittableGenerator` interface.
java.util.SplittableRandom.splits(long, RandomGenerator.SplittableGenerator)	17	Returns a stream producing the given `streamSize` number of new pseudorandom number generators, each of which implements the `RandomGenerator.SplittableGenerator` interface.
java.util.SplittableRandom.splits (RandomGenerator.SplittableGenerator)	17	Returns an effectively unlimited stream of new pseudorandom number generators, each of which implements the `RandomGenerator.SplittableGenerator` interface.
java.util.stream.Collectors.teeing(Collector<? super T, ?, R1>, Collector<? super T, ?, R2>, BiFunction<? super R1, ? super R2, R>)	12	Returns a `Collector` that is a composite of two downstream collectors.
java.util.stream.DoubleStream.mapMulti (DoubleStream.DoubleMapMultiConsumer)	16	Returns a stream consisting of the results of replacing each element of this stream with multiple elements, specifically zero or more elements.
java.util.stream.IntStream.mapMulti (IntStream.IntMapMultiConsumer)	16	Returns a stream consisting of the results of replacing each element of this stream with multiple elements, specifically zero or more elements.
java.util.stream.LongStream.mapMulti (LongStream.LongMapMultiConsumer)	16	Returns a stream consisting of the results of replacing each element of this stream with multiple elements, specifically zero or more elements.
java.util.stream.Stream.mapMulti(BiConsumer<? super T, ? super Consumer<R>>)	16	Returns a stream consisting of the results of replacing each element of this stream with multiple elements, specifically zero or more elements.
java.util.stream.Stream.mapMultiToDouble (BiConsumer<? super T, ? super DoubleConsumer>)	16	Returns a `DoubleStream` consisting of the results of replacing each element of this stream with multiple elements, specifically zero or more elements.
java.util.stream.Stream.mapMultiToInt(BiConsumer<? super T, ? super IntConsumer>)	16	Returns an `IntStream` consisting of the results of replacing each element of this stream with multiple elements, specifically zero or more elements.
java.util.stream.Stream.mapMultiToLong(BiConsumer<? super T, ? super LongConsumer>)	16	Returns a `LongStream` consisting of the results of replacing each element of this stream with multiple elements, specifically zero or more elements.
java.util.stream.Stream.toList()	16	Accumulates the elements of this stream into a `List`.

java.util.TreeMap.putFirst(K, V)	21	Throws `UnsupportedOperationException`.
java.util.TreeMap.putLast(K, V)	21	Throws `UnsupportedOperationException`.
java.util.TreeSet.addFirst(E)	21	Throws `UnsupportedOperationException`.
java.util.TreeSet.addLast(E)	21	Throws `UnsupportedOperationException`.
java.util.WeakHashMap.newWeakHashMap(int)	19	Creates a new, empty WeakHashMap suitable for the expected number of mappings.
java.util.zip.ZipInputStream.skipNBytes(long)	12	Skips over and discards exactly n bytes of data from this input stream for the current ZIP entry.
javax.annotation.processing.Messager.printError (CharSequence)	18	Prints an error.
javax.annotation.processing.Messager.printError (CharSequence, Element)	18	Prints an error at the location of the element.
javax.annotation.processing.Messager.printNote (CharSequence)	18	Prints a note.
javax.annotation.processing.Messager.printNote (CharSequence, Element)	18	Prints a note at the location of the element.
javax.annotation.processing.Messager.printWarning (CharSequence)	18	Prints a warning.
javax.annotation.processing.Messager.printWarning (CharSequence, Element)	18	Prints a warning at the location of the element.
javax.annotation.processing.ProcessingEnvironment.isP	13	Returns `true` if *preview features* are enabled and `false` otherwise.
javax.lang.model.element.ElementKind.isDeclaredType()	19	Returns `true` if this is a kind of declared type, a class or an interface, and `false` otherwise.
javax.lang.model.element.ElementKind.isExecutable()	19	Returns `true` if this is a kind of executable: either `METHOD` or `CONSTRUCTOR` or `STATIC_INIT` or `INSTANCE_INIT`.
javax.lang.model.element.ElementKind.isInitializer()	19	Returns `true` if this is a kind of initializer: either `STATIC_INIT` or `INSTANCE_INIT`.
javax.lang.model.element.ElementKind.isVariable()	19	Returns `true` if this is a kind of variable: including `ENUM_CONSTANT`, `FIELD`, `PARAMETER`, `LOCAL_VARIABLE`, `EXCEPTION_PARAMETER`, `RESOURCE_VARIABLE`, and `BINDING_VARIABLE`.
javax.lang.model.element.ElementVisitor.visitRecordCom (RecordComponentElement, P)	16	Visits a record component element.
javax.lang.model.element.TypeElement.getPermittedSub	17	Returns the permitted classes of this class or interface element in declaration order.
javax.lang.model.element.TypeElement.getRecordCompo	16	Returns the record components of this class or interface element in declaration order.
javax.lang.model.element.TypeElement.isUnnamed()^{PREVIE}	21	Returns `true` if this is an unnamed class and `false` otherwise.
javax.lang.model.element.VariableElement.isUnnamed()	21	Returns `true` if this is an unnamed variable and `false` otherwise.
javax.lang.model.SourceVersion.runtimeVersion()	18	Returns the least runtime version that supports this source version; otherwise `null`.
javax.lang.model.SourceVersion.valueOf (Runtime.Version)	18	Returns the latest source version that is usable under the runtime version argument.
javax.lang.model.util.AbstractElementVisitor6.visitRecor (RecordComponentElement, P)	14	Visits a record component element.
javax.lang.model.util.ElementFilter.recordComponentsIn (Iterable<? extends Element>)	16	Returns a list of record components in `elements`.
javax.lang.model.util.ElementFilter.recordComponentsIn (Set<? extends Element>)	16	Returns a set of record components in `elements`.

javax.lang.model.util.ElementKindVisitor14.visitVariable (VariableElement, P)	14	Visits a `BINDING_VARIABLE` variable element.
javax.lang.model.util.ElementKindVisitor6.visitTypeAsRe (TypeElement, P)	16	Visits a `RECORD` type element.
javax.lang.model.util.ElementKindVisitor6.visitVariableA (VariableElement, P)	14	Visits a `BINDING_VARIABLE` variable element.
javax.lang.model.util.Elements.getFileObjectOf (Element)	18	Returns the file object for this element or `null` if there is no such file object.
javax.lang.model.util.Elements.getOutermostTypeElemen (Element)	18	Returns the outermost type element an element is contained in if such a containing element exists; otherwise returns `null`.
javax.lang.model.util.Elements.isAutomaticModule (ModuleElement)	17	Returns `true` if the module element is an automatic module, `false` otherwise.
javax.lang.model.util.Elements.isCanonicalConstructor (ExecutableElement)	20	Returns `true` if the executable element can be determined to be a canonical constructor of a record, `false` otherwise.
javax.lang.model.util.Elements.isCompactConstructor (ExecutableElement)	20	Returns `true` if the executable element can be determined to be a compact constructor of a record, `false` otherwise.
javax.lang.model.util.Elements.recordComponentFor (ExecutableElement)	16	Returns the record component for the given accessor.
javax.net.ssl.HttpsURLConnection.getSSLSession()	12	Returns an `Optional` containing the `SSLSession` in use on this connection.
javax.net.ssl.SSLParameters.getNamedGroups()	20	Returns a prioritized array of key exchange named groups names that can be used over the SSL/TLS/DTLS protocols.
javax.net.ssl.SSLParameters.getSignatureSchemes()	19	Returns a prioritized array of signature scheme names that can be used over the SSL/TLS/DTLS protocols.
javax.net.ssl.SSLParameters.setNamedGroups(String[])	20	Sets the prioritized array of key exchange named groups names that can be used over the SSL/TLS/DTLS protocols.
javax.net.ssl.SSLParameters.setSignatureSchemes (String[])	19	Sets the prioritized array of signature scheme names that can be used over the SSL/TLS/DTLS protocols.
javax.security.auth.Subject.callAs(Subject, Callable<T>)	18	Executes a `Callable` with `subject` as the current subject.
javax.security.auth.Subject.current()	18	Returns the current subject.
javax.swing.filechooser.FileSystemView.getChooserShort	12	Returns an array of files representing the values to show by default in the file chooser shortcuts panel.
javax.swing.filechooser.FileSystemView.getSystemIcon (File, int, int)	17	Returns an icon for a file, directory, or folder as it would be displayed in a system file browser for the requested size.
javax.tools.JavaFileManager.getFileForOutputForOriginal (JavaFileManager.Location, String, String, FileObject...)	18	Returns a file object for output representing the specified relative name in the specified package in the given location.
javax.tools.JavaFileManager.getJavaFileForOutputForOrig (JavaFileManager.Location, String, JavaFileObject.Kind, FileObject...)	18	Returns a file object for output representing the specified class of the specified kind in the given package-oriented location.
javax.tools.StandardJavaFileManager.getJavaFileObjects (Collection<? extends Path>)	13	Returns file objects representing the given paths.
javax.xml.parsers.DocumentBuilderFactory.newDefaultN	13	Creates a new NamespaceAware instance of the `DocumentBuilderFactory` builtin system-default implementation.
javax.xml.parsers.DocumentBuilderFactory.newNSInstan	13	Creates a new NamespaceAware instance of a `DocumentBuilderFactory`.

API	Ver	Description
javax.xml.parsers.DocumentBuilderFactory.newNSInstan (String, ClassLoader)	13	Creates a new NamespaceAware instance of a `DocumentBuilderFactory` from the class name.
javax.xml.parsers.SAXParserFactory.newDefaultNSInstar	13	Creates a new NamespaceAware instance of the `SAXParserFactory` builtin system-default implementation.
javax.xml.parsers.SAXParserFactory.newNSInstance()	13	Creates a new NamespaceAware instance of a `SAXParserFactory`.
javax.xml.parsers.SAXParserFactory.newNSInstance (String, ClassLoader)	13	Creates a new NamespaceAware instance of a `SAXParserFactory` from the class name.
javax.xml.xpath.XPathFactory.getProperty(String)	18	Returns the value of the specified property.
javax.xml.xpath.XPathFactory.setProperty(String, String)	18	Sets a property for this `XPathFactory`.
jdk.incubator.vector.ByteVector.compress (VectorMask<Byte>)	19	Compresses the lane elements of this vector selecting lanes under the control of a specific mask.
jdk.incubator.vector.ByteVector.expand (VectorMask<Byte>)	19	Expands the lane elements of this vector under the control of a specific mask.
jdk.incubator.vector.ByteVector.fromMemorySegment (VectorSpecies<Byte>, MemorySegment, long, ByteOrder)[PREVIEW]	19	Loads a vector from a memory segment[PREVIEW] starting at an offset into the memory segment.
jdk.incubator.vector.ByteVector.fromMemorySegment (VectorSpecies<Byte>, MemorySegment, long, ByteOrder, VectorMask<Byte>)[PREVIEW]	19	Loads a vector from a memory segment[PREVIEW] starting at an offset into the memory segment and using a mask.
jdk.incubator.vector.ByteVector.intoMemorySegment (MemorySegment, long, ByteOrder)[PREVIEW]	19	Stores this vector into a memory segment[PREVIEW] starting at an offset using explicit byte order.
jdk.incubator.vector.ByteVector.intoMemorySegment (MemorySegment, long, ByteOrder, VectorMask<Byte>)[PREVIEW]	19	Stores this vector into a memory segment[PREVIEW] starting at an offset using explicit byte order and a mask.
jdk.incubator.vector.DoubleVector.compress (VectorMask<Double>)	19	Compresses the lane elements of this vector selecting lanes under the control of a specific mask.
jdk.incubator.vector.DoubleVector.expand (VectorMask<Double>)	19	Expands the lane elements of this vector under the control of a specific mask.
jdk.incubator.vector.DoubleVector.fromMemorySegment (VectorSpecies<Double>, MemorySegment, long, ByteOrder)[PREVIEW]	19	Loads a vector from a memory segment[PREVIEW] starting at an offset into the memory segment.
jdk.incubator.vector.DoubleVector.fromMemorySegment (VectorSpecies<Double>, MemorySegment, long, ByteOrder, VectorMask<Double>)[PREVIEW]	19	Loads a vector from a memory segment[PREVIEW] starting at an offset into the memory segment and using a mask.
jdk.incubator.vector.DoubleVector.intoMemorySegment (MemorySegment, long, ByteOrder)[PREVIEW]	19	Stores this vector into a memory segment[PREVIEW] starting at an offset using explicit byte order.
jdk.incubator.vector.DoubleVector.intoMemorySegment (MemorySegment, long, ByteOrder, VectorMask<Double>)[PREVIEW]	19	Stores this vector into a memory segment[PREVIEW] starting at an offset using explicit byte order and a mask.
jdk.incubator.vector.FloatVector.compress (VectorMask<Float>)	19	Compresses the lane elements of this vector selecting lanes under the control of a specific mask.
jdk.incubator.vector.FloatVector.expand (VectorMask<Float>)	19	Expands the lane elements of this vector under the control of a specific mask.
jdk.incubator.vector.FloatVector.fromMemorySegment (VectorSpecies<Float>, MemorySegment, long, ByteOrder)[PREVIEW]	19	Loads a vector from a memory segment[PREVIEW] starting at an offset into the memory segment.
jdk.incubator.vector.FloatVector.fromMemorySegment (VectorSpecies<Float>, MemorySegment, long, ByteOrder, VectorMask<Float>)[PREVIEW]	19	Loads a vector from a memory segment[PREVIEW] starting at an offset into the memory segment and using a mask.
jdk.incubator.vector.FloatVector.intoMemorySegment (MemorySegment, long, ByteOrder)[PREVIEW]	19	Stores this vector into a memory segment[PREVIEW] starting at an offset using explicit byte order.
jdk.incubator.vector.FloatVector.intoMemorySegment (MemorySegment, long, ByteOrder, VectorMask<Float>)[PREVIEW]	19	Stores this vector into a memory segment[PREVIEW] starting at an offset using explicit byte order and a mask.
jdk.incubator.vector.IntVector.compress (VectorMask<Integer>)	19	Compresses the lane elements of this vector selecting lanes under the control of a specific mask.
jdk.incubator.vector.IntVector.expand (VectorMask<Integer>)	19	Expands the lane elements of this vector under the control of a specific mask.

jdk.incubator.vector.IntVector.fromMemorySegment (VectorSpecies\<Integer>, MemorySegment, long, ByteOrder)^{PREVIEW}	19	Loads a vector from a memory segment^{PREVIEW} starting at an offset into the memory segment.
jdk.incubator.vector.IntVector.fromMemorySegment (VectorSpecies\<Integer>, MemorySegment, long, ByteOrder, VectorMask\<Integer>)^{PREVIEW}	19	Loads a vector from a memory segment^{PREVIEW} starting at an offset into the memory segment and using a mask.
jdk.incubator.vector.IntVector.intoMemorySegment (MemorySegment, long, ByteOrder)^{PREVIEW}	19	Stores this vector into a memory segment^{PREVIEW} starting at an offset using explicit byte order.
jdk.incubator.vector.IntVector.intoMemorySegment (MemorySegment, long, ByteOrder, VectorMask\<Integer>)^{PREVIEW}	19	Stores this vector into a memory segment^{PREVIEW} starting at an offset using explicit byte order and a mask.
jdk.incubator.vector.LongVector.compress (VectorMask\<Long>)	19	Compresses the lane elements of this vector selecting lanes under the control of a specific mask.
jdk.incubator.vector.LongVector.expand (VectorMask\<Long>)	19	Expands the lane elements of this vector under the control of a specific mask.
jdk.incubator.vector.LongVector.fromMemorySegment (VectorSpecies\<Long>, MemorySegment, long, ByteOrder)^{PREVIEW}	19	Loads a vector from a memory segment^{PREVIEW} starting at an offset into the memory segment.
jdk.incubator.vector.LongVector.fromMemorySegment (VectorSpecies\<Long>, MemorySegment, long, ByteOrder, VectorMask\<Long>)^{PREVIEW}	19	Loads a vector from a memory segment^{PREVIEW} starting at an offset into the memory segment and using a mask.
jdk.incubator.vector.LongVector.intoMemorySegment (MemorySegment, long, ByteOrder)^{PREVIEW}	19	Stores this vector into a memory segment^{PREVIEW} starting at an offset using explicit byte order.
jdk.incubator.vector.LongVector.intoMemorySegment (MemorySegment, long, ByteOrder, VectorMask\<Long>)^{PREVIEW}	19	Stores this vector into a memory segment^{PREVIEW} starting at an offset using explicit byte order and a mask.
jdk.incubator.vector.ShortVector.compress (VectorMask\<Short>)	19	Compresses the lane elements of this vector selecting lanes under the control of a specific mask.
jdk.incubator.vector.ShortVector.expand (VectorMask\<Short>)	19	Expands the lane elements of this vector under the control of a specific mask.
jdk.incubator.vector.ShortVector.fromMemorySegment (VectorSpecies\<Short>, MemorySegment, long, ByteOrder)^{PREVIEW}	19	Loads a vector from a memory segment^{PREVIEW} starting at an offset into the memory segment.
jdk.incubator.vector.ShortVector.fromMemorySegment (VectorSpecies\<Short>, MemorySegment, long, ByteOrder, VectorMask\<Short>)^{PREVIEW}	19	Loads a vector from a memory segment^{PREVIEW} starting at an offset into the memory segment and using a mask.
jdk.incubator.vector.ShortVector.intoMemorySegment (MemorySegment, long, ByteOrder)^{PREVIEW}	19	Stores this vector into a memory segment^{PREVIEW} starting at an offset using explicit byte order.
jdk.incubator.vector.ShortVector.intoMemorySegment (MemorySegment, long, ByteOrder, VectorMask\<Short>)^{PREVIEW}	19	Stores this vector into a memory segment^{PREVIEW} starting at an offset using explicit byte order and a mask.
jdk.incubator.vector.Vector.compress(VectorMask\<E>)	19	Compresses the lane elements of this vector selecting lanes under the control of a specific mask.
jdk.incubator.vector.Vector.expand(VectorMask\<E>)	19	Expands the lane elements of this vector under the control of a specific mask.
jdk.incubator.vector.Vector.intoMemorySegment (MemorySegment, long, ByteOrder)^{PREVIEW}	19	Stores this vector into a memory segment^{PREVIEW} starting at an offset using explicit byte order.
jdk.incubator.vector.Vector.intoMemorySegment (MemorySegment, long, ByteOrder, VectorMask\<E>)^{PREVIEW}	19	Stores this vector into a memory segment^{PREVIEW} starting at an offset using explicit byte order and a mask.
jdk.incubator.vector.VectorMask.compress()	19	Compresses set lanes from this mask.
jdk.incubator.vector.VectorMask.indexInRange(long, long)	19	Removes lanes numbered N from this mask where the adjusted index N+offset, is not in the range [0..limit-1].
jdk.incubator.vector.VectorSpecies.fromMemorySegment (MemorySegment, long, ByteOrder)^{PREVIEW}	19	Loads a vector of this species from a memory segment^{PREVIEW} starting at an offset into the memory segment.
jdk.incubator.vector.VectorSpecies.indexInRange(long, long)	19	Returns a mask of this species where only the lanes at index N such that the adjusted index N+offset is in the range [0..limit-1] are set.
jdk.incubator.vector.VectorSpecies.loopBound(long)	19	Loop control function which returns the largest multiple of VLENGTH that is less than or equal to the given length value.

jdk.javadoc.doclet.Reporter.getDiagnosticWriter()	17	Returns a writer that can be used to write diagnostic output, or `null` if no such writer is available.
jdk.javadoc.doclet.Reporter.getStandardWriter()	17	Returns a writer that can be used to write non-diagnostic output, or `null` if no such writer is available.
jdk.javadoc.doclet.Reporter.print(Diagnostic.Kind, DocTreePath, int, int, int, String)	18	Prints a diagnostic message related to a position within a range of characters in a tree node.
jdk.javadoc.doclet.Reporter.print(Diagnostic.Kind, FileObject, int, int, int, String)	17	Prints a diagnostic message related to a position within a range of characters in a file.
jdk.javadoc.doclet.StandardDoclet.getLocale()	17	Returns the locale for this doclet.
jdk.javadoc.doclet.StandardDoclet.getReporter()	17	Returns the reporter for this doclet.
jdk.jfr.consumer.EventStream.onMetadata (Consumer<MetadataEvent>)	16	Registers an action to perform when new metadata arrives in the stream.
jdk.jfr.consumer.RecordedThread.isVirtual()	21	Returns `true` if this is a virtual Thread, `false` otherwise.
jdk.jfr.consumer.RecordingFile.write(Path, Predicate<RecordedEvent>)	19	Filter out events and write them to a new file.
jdk.jfr.consumer.RecordingStream.dump(Path)	17	Writes recording data to a file.
jdk.jfr.consumer.RecordingStream.stop()	20	Stops the recording stream.
jdk.jshell.JShell.Builder.console(JShellConsole)	21	Sets the console for the running evaluation.
jdk.jshell.SourceCodeAnalysis.highlights(String)	19	Returns a collection of `Highlights` which can be used to color the given snippet.
jdk.jshell.spi.ExecutionEnv.console()	21	Returns the `JShellConsole` that should be used by the execution engine, or `null` if none.
jdk.jshell.tool.JavaShellToolBuilder.interactiveTerminal (boolean)	17	Set to true to specify the inputs and outputs are connected to an interactive terminal that can interpret the ANSI escape codes.
jdk.management.jfr.RemoteRecordingStream.dump (Path)	17	Writes recording data to a file.
jdk.management.jfr.RemoteRecordingStream.stop()	20	Stops the recording stream.
org.xml.sax.ContentHandler.declaration(String, String, String)	14	Receives notification of the XML declaration.

New Constructors

Constructor ⬍	Added in ⬍	Description
com.sun.net.httpserver.Headers(Map<String, List<String>>)	18	Creates a mutable `Headers` from the given `headers` with the same header names and values.
java.io.InvalidClassException(String, String, Throwable)	19	Report an InvalidClassException for the reason and cause specified.
java.io.InvalidClassException(String, Throwable)	19	Report an InvalidClassException for the reason and cause specified.
java.io.InvalidObjectException(String, Throwable)	19	Constructs an `InvalidObjectException` with the given reason and cause.
java.io.ObjectStreamException(String, Throwable)	19	Create an ObjectStreamException with the specified message and cause.
java.io.ObjectStreamException(Throwable)	19	Create an ObjectStreamException with the specified cause.
java.lang.IndexOutOfBoundsException(long)	16	Constructs a new `IndexOutOfBoundsException` class with an argument indicating the illegal index.
java.net.ServerSocket(SocketImpl)	12	Creates a server socket with a user-specified `SocketImpl`.
java.net.SocketException(String, Throwable)	19	Constructs a new `SocketException` with the specified detail message and cause.
java.net.SocketException(Throwable)	19	Constructs a new `SocketException` with the specified cause.

java.security.InvalidParameterException(String, Throwable)	20	Constructs an InvalidParameterException with the specified detail message and cause.
java.security.InvalidParameterException(Throwable)	20	Constructs an InvalidParameterException with the specified cause and a detail message of (cause==null ? null : cause.toString()) (which typically contains the class and detail message of cause).
java.text.CompactNumberFormat(String, DecimalFormatSymbols, String[], String)	14	Creates a CompactNumberFormat using the given decimal pattern, decimal format symbols, compact patterns, and plural rules.
java.util.concurrent.ForkJoinWorkerThread (ThreadGroup, ForkJoinPool, boolean)	19	Creates a ForkJoinWorkerThread operating in the given thread group and pool, and with the given policy for preserving ThreadLocals.
java.util.NoSuchElementException(String, Throwable)	15	Constructs a NoSuchElementException with the specified detail message and cause.
java.util.NoSuchElementException(Throwable)	15	Constructs a NoSuchElementException with the specified cause.
javax.net.ssl.SSLHandshakeException(String, Throwable)	19	Creates a SSLHandshakeException with the specified detail message and cause.
javax.net.ssl.SSLKeyException(String, Throwable)	19	Creates a SSLKeyException with the specified detail message and cause.
javax.net.ssl.SSLPeerUnverifiedException(String, Throwable)	19	Creates a SSLPeerUnverifiedException with the specified detail message and cause.
javax.net.ssl.SSLProtocolException(String, Throwable)	19	Creates a SSLProtocolException with the specified detail message and cause.

New Enum Constants

Enum Constant	Added in	Description
com.sun.source.doctree.DocTree.Kind.ESCAPE	21	Used for instances of EscapeTree representing some escaped documentation text.
com.sun.source.doctree.DocTree.Kind.SNIPPET	18	Used for instances of SnippetTree representing an @snippet tag.
com.sun.source.doctree.DocTree.Kind.SPEC	20	Used for instances of SpecTree representing an @spec tag.
com.sun.source.doctree.DocTree.Kind.SYSTEM_PROPERT	12	Used for instances of SystemPropertyTree representing an @systemProperty tag.
com.sun.source.tree.Tree.Kind.ANY_PATTERN[PREVIEW]	21	Used for instances of BindingPatternTree.
com.sun.source.tree.Tree.Kind.BINDING_PATTERN	16	Used for instances of BindingPatternTree.
com.sun.source.tree.Tree.Kind.CONSTANT_CASE_LABEL	21	Used for instances of ConstantCaseLabelTree.
com.sun.source.tree.Tree.Kind.DECONSTRUCTION_PATTE	21	Used for instances of DeconstructionPatternTree.
com.sun.source.tree.Tree.Kind.DEFAULT_CASE_LABEL	21	Used for instances of DefaultCaseLabelTree.
com.sun.source.tree.Tree.Kind.PATTERN_CASE_LABEL	21	Used for instances of PatternCaseLabelTree.
com.sun.source.tree.Tree.Kind.RECORD	16	Used for instances of ClassTree representing records.
com.sun.source.tree.Tree.Kind.SWITCH_EXPRESSION	12	Used for instances of SwitchExpressionTree.
com.sun.source.tree.Tree.Kind.YIELD	13	Used for instances of YieldTree.
java.lang.annotation.ElementType.RECORD_COMPONENT	16	Record component
java.lang.Character.UnicodeScript.CHORASMIAN	15	Unicode script "Chorasmian".
java.lang.Character.UnicodeScript.CYPRO_MINOAN	19	Unicode script "Cypro Minoan".
java.lang.Character.UnicodeScript.DIVES_AKURU	15	Unicode script "Dives Akuru".
java.lang.Character.UnicodeScript.DOGRA	12	Unicode script "Dogra".
java.lang.Character.UnicodeScript.ELYMAIC	13	Unicode script "Elymaic".

java.lang.Character.UnicodeScript.GUNJALA_GONDI	12	Unicode script "Gunjala Gondi".
java.lang.Character.UnicodeScript.HANIFI_ROHINGYA	12	Unicode script "Hanifi Rohingya".
java.lang.Character.UnicodeScript.KAWI	20	Unicode script "Kawi".
java.lang.Character.UnicodeScript.KHITAN_SMALL_SCRIPT	15	Unicode script "Khitan Small Script".
java.lang.Character.UnicodeScript.MAKASAR	12	Unicode script "Makasar".
java.lang.Character.UnicodeScript.MEDEFAIDRIN	12	Unicode script "Medefaidrin".
java.lang.Character.UnicodeScript.NAG_MUNDARI	20	Unicode script "Nag Mundari".
java.lang.Character.UnicodeScript.NANDINAGARI	13	Unicode script "Nandinagari".
java.lang.Character.UnicodeScript.NYIAKENG_PUACHUE_HMONG	13	Unicode script "Nyiakeng Puachue Hmong".
java.lang.Character.UnicodeScript.OLD_SOGDIAN	12	Unicode script "Old Sogdian".
java.lang.Character.UnicodeScript.OLD_UYGHUR	19	Unicode script "Old Uyghur".
java.lang.Character.UnicodeScript.SOGDIAN	12	Unicode script "Sogdian".
java.lang.Character.UnicodeScript.TANGSA	19	Unicode script "Tangsa".
java.lang.Character.UnicodeScript.TOTO	19	Unicode script "Toto".
java.lang.Character.UnicodeScript.VITHKUQI	19	Unicode script "Vithkuqi".
java.lang.Character.UnicodeScript.WANCHO	13	Unicode script "Wancho".
java.lang.Character.UnicodeScript.YEZIDI	15	Unicode script "Yezidi".
java.lang.reflect.ClassFileFormatVersion.RELEASE_21	21	The version introduced by the Java Platform, Standard Edition 21.
java.net.StandardProtocolFamily.UNIX	16	Unix domain (Local) interprocess communication.
javax.lang.model.element.ElementKind.BINDING_VARIABLE	16	A binding variable in a pattern.
javax.lang.model.element.ElementKind.RECORD	16	A record class.
javax.lang.model.element.ElementKind.RECORD_COMPONENT	16	A record component of a `record`.
javax.lang.model.element.Modifier.NON_SEALED	17	The modifier `non-sealed`
javax.lang.model.element.Modifier.SEALED	17	The modifier `sealed`
javax.lang.model.SourceVersion.RELEASE_12	12	The version introduced by the Java Platform, Standard Edition 12.
javax.lang.model.SourceVersion.RELEASE_13	13	The version introduced by the Java Platform, Standard Edition 13.
javax.lang.model.SourceVersion.RELEASE_14	14	The version introduced by the Java Platform, Standard Edition 14.
javax.lang.model.SourceVersion.RELEASE_15	15	The version introduced by the Java Platform, Standard Edition 15.
javax.lang.model.SourceVersion.RELEASE_16	16	The version introduced by the Java Platform, Standard Edition 16.
javax.lang.model.SourceVersion.RELEASE_17	17	The version introduced by the Java Platform, Standard Edition 17.
javax.lang.model.SourceVersion.RELEASE_18	18	The version introduced by the Java Platform, Standard Edition 18.
javax.lang.model.SourceVersion.RELEASE_19	19	The version introduced by the Java Platform, Standard Edition 19.
javax.lang.model.SourceVersion.RELEASE_20	20	The version introduced by the Java Platform, Standard Edition 20.
javax.lang.model.SourceVersion.RELEASE_21	21	The version introduced by the Java Platform, Standard Edition 21.
jdk.jshell.Snippet.SubKind.RECORD_SUBKIND	14	A record declaration.