## **Java Keywords**

The following 51 keywords cannot be used as identifiers.

| **Keyword** | **Description** |
| --- | --- |
| [*abstract*](https://howtodoinjava.com/java/keywords/abstract-keyword/) | Used with classes and methods. An *abstract class* cannot be instantiated. An *abstract method* is incomplete without the body and must be implemented in the child class to create an instance of the child class. |
| [*assert*](https://howtodoinjava.com/java/keywords/java-assert/) | enables us to test the assumptions about our program. |
| [*boolean*](https://howtodoinjava.com/java/keywords/java-boolean/) | represents only one of two possible values i.e. either true or false. |
| [*break*](https://howtodoinjava.com/java/flow-control/break-keyword/) | is used to terminate *for*, *while*, or *do-while* loop. It may also be used to terminate a *switch* statement as well. |
| *byte* | can store whole numbers from -128 and 127. |
| [*case*](https://howtodoinjava.com/java/flow-control/switch-statement-in-java/) | represents a block of code in *switch* statements. |
| [*catch*](https://howtodoinjava.com/java/exception-handling/try-catch-finally/) | follows the try block and handles the checked exceptions thrown by *try* block and any possible unchecked exceptions. |
| *char* | used to store a single character. |
| *class* | defines a class. |
| *const* | is a reserved keyword for constant values. Use *final* instead. |
| [*continue*](https://howtodoinjava.com/java/flow-control/continue-keyword-statement-in-java/) | skips the current iteration of a *for*, *while*, or *do-while* loops and jumps to the next iteration. |
| [*default*](https://howtodoinjava.com/java8/default-methods-in-java-8/) | used to specify the default block in a *switch* statement and *default methods* in functional interfaces. |
| [*do*](https://howtodoinjava.com/java/flow-control/do-while-loop-in-java/) | used to contain the statements to execute repeatedly until the condition in the *while* statement is *true*. |
| *double* | used to declare a variable that can hold 64-bit floating-point number. |
| [*else*](https://howtodoinjava.com/java/flow-control/if-else-statement-in-java/) | used to indicate the alternative branches in an *if* statement. |
| [*enum*](https://howtodoinjava.com/java/enum/enum-tutorial/) | is a type whose fields consist of a fixed set of constants. |
| [*extends*](https://howtodoinjava.com/java/oops/extends-vs-implements/) | used for extending a class. |
| [*final*](https://howtodoinjava.com/java/keywords/final-finally-finalize/) | used with class variables, methods or classes. A *final* variable cannot be assigned another value after it has been initialized. A *final* method cannot be overridden in the child class. No class can subclass a *final* class. |
| [*finally*](https://howtodoinjava.com/java/exception-handling/try-catch-finally/) | contains code to be executed everytime a try-catch block is completed – either with errors or without any error. |
| *float* | used to declare a variable that can hold a 32-bit floating-point number. |
| [*for*](https://howtodoinjava.com/java/flow-control/for-loop-in-java/) | start a loop to execute a set of instructions repeatedly when a condition is *true*. If the number of iterations is known, it is recommended to use *for* loop. |
| *goto* | Currently, not in use. |
| [*if*](https://howtodoinjava.com/java/flow-control/if-else-statement-in-java/) | used for writing conditional statements. |
| [*implements*](https://howtodoinjava.com/java/oops/extends-vs-implements/) | used for implementing an interface. |
| *import* | import a package, class or interface to the current class. |
| [*instanceof*](https://howtodoinjava.com/java/oops/java-instanceof/) | Checks whether an object is an instance of a specific class or an interface. |
| *int* | used to store a 32-bit integer value. |
| *interface* | declares an interface. |
| *long* | used to store a 64-bit integer value. |
| *native* | indicates native code (platform-specific). |
| *new* | creates a new object of the specified class. |
| *package* | declares a package for storing the related classes. |
| [*private*](https://howtodoinjava.com/java/oops/java-access-modifiers/) | access modifier to indicate that a method or variable may be accessed only in the class in which it is declared. |
| [*protected*](https://howtodoinjava.com/java/oops/java-access-modifiers/) | access modifier to indicate that a class, method or variable may be accessed only in the current package, or inherited outside the current package. |
| [*public*](https://howtodoinjava.com/java/oops/java-access-modifiers/) | access modifier to indicate that a class, method or variable is accessible everywhere. |
| *return* | used to return from a method when its execution is complete. |
| *short* | used to store a 16-bit integer value. |
| [*static*](https://howtodoinjava.com/java/keywords/java-static-keyword/) | indicates that a variable or method belongs to the *class* object, not to the individual instances of that class. |
| [*strictfp*](https://howtodoinjava.com/java/keywords/strictfp-modifier/) | used to restrict the floating-point calculations to ensure portability. |
| [*super*](https://howtodoinjava.com/java/keywords/this-vs-super/) | used to refer to parent class objects. |
| [*switch*](https://howtodoinjava.com/java/flow-control/switch-statement-in-java/) | help in providing multiple possible execution paths for a program. |
| [*synchronized*](https://howtodoinjava.com/java/keywords/java-synchronized/) | marks a block or method a critical section where one and only one thread is executing at a time. |
| [*this*](https://howtodoinjava.com/java/keywords/this-vs-super/) | used to refer to the current object. |
| [*throw*](https://howtodoinjava.com/java/exception-handling/throw-vs-throws/) | used to explicitly throw an exception from a method or constructor. |
| [*throws*](https://howtodoinjava.com/java/exception-handling/throw-vs-throws/) | used to declare the list of exceptions that may be thrown by that method or constructor. |
| [*transient*](https://howtodoinjava.com/java/keywords/transient-keyword-in-java-with-real-time-example/) | used on class attributes/variables to indicate that the serialization process of this class should ignore such variables. |
| [*try*](https://howtodoinjava.com/java/exception-handling/try-catch-finally/) | contains the application code which is expected to work in normal conditions. |
| *void* | specifies that a method should not have a return value. |
| *volatile* | indicates that an attribute is not cached thread-locally, and is always read from the “main memory”. |
| [*while*](https://howtodoinjava.com/java/flow-control/while-loop-in-java/) | continually executes a block of statements until a particular condition evaluates to true |
| \_ (*Underscore*) | added in Java 9, to prevent writing underscores as an unused lambda, method, or catch formal parameter. |

## **3. Contextual Keywords**

The following 16 words can be interpreted as keywords or as other tokens, depending on the context in which they appear.

| **Keyword** | **Description** |
| --- | --- |
| *exports* | used for importing and exporting the modules. |
| [*module*](https://howtodoinjava.com/java9/java-9-modules-tutorial/) | used for declaring modules. |
| *non-sealed* | used to define sealed classes and interfaces. |
| *open* | used for declaring modules. |
| *opens* | used for importing and exporting the modules. |
| *permits* | used to define sealed classes and interfaces. |
| *provides* | used for importing and exporting the modules. |
| [*record*](https://howtodoinjava.com/java/java-record-type/) | used to define new |
| *requires* | used for importing and exporting the modules. |
| [*sealed*](https://howtodoinjava.com/java15/sealed-classes-interfaces/) | used to define sealed classes and interfaces. |
| *to* | used for importing and exporting the modules. |
| *transitive* | recognized as a terminal in a *RequiresModifier*. |
| *uses* | used for importing and exporting the modules. |
| [*var*](https://howtodoinjava.com/java10/var-local-variable-type-inference/) | used to infer local variable types. |
| *with* | used for importing and exporting the modules. |
| [*yield*](https://howtodoinjava.com/java/keywords/yield-keyword-in-java/) | used to yield a value in a *switch* statement. |

**Before Java 10:**

java

CopyEdit

String name = "Mahi";

int age = 25;

List<String> names = new ArrayList<>();

**With var:**

java

CopyEdit

var name = "Mahi"; // Inferred as String

var age = 31; // Inferred as int

var names = new ArrayList<String>(); // Inferred as ArrayList<String>

int code = 404;

String message = switch (code) {

case 200 -> "OK";

case int i when i >= 100 && i < 200 -> "Informational";

case int i when i >= 200 && i < 300 -> "Success";

case int i when i >= 400 && i < 500 -> "Client error";

default -> "Other";

};

System.out.println(message); // prints "Client error"