



## Java Keywords

Keyword	Definition
<b>abstract</b>	A Java(TM) programming language keyword used in a class definition to specify that a class is not to be instantiated, but rather inherited by other classes. An <b>abstract</b> class can have <b>abstract</b> methods that are not implemented in the <b>abstract</b> class, but in subclasses.
<b>boolean</b>	Refers to an expression or variable that can have only a true or false value. The Java(TM) programming language provides the <b>boolean</b> type and the literal values true and false.
<b>break</b>	A Java(TM) programming language keyword used to resume program execution at the statement immediately following the current statement. If followed by a label, the program resumes execution at the labeled statement.
<b>byte</b>	A sequence of eight bits. The Java(TM) programming language provides a corresponding byte type.
<b>case</b>	A Java(TM) programming language keyword that defines a group of statements to begin executing if a value specified matches the value defined by a preceding <b>switch</b> keyword.
<b>catch</b>	A Java(TM) programming language keyword used to declare a block of statements to be executed in the event that a Java exception, or runtime error, occurs in a preceding <b>try</b> block.
<b>char</b>	A Java(TM) programming language keyword used to declare a variable of type character.
<b>class</b>	In the Java(TM) programming language, a type that defines the implementation of a particular kind of object. A <b>class</b> definition defines instance and <b>class</b> variables and methods, as well as specifying the interfaces the <b>class</b> implements and the immediate superclass of the <b>class</b> . If the superclass is not explicitly specified, the superclass will implicitly be <b>object</b> .
<b>const</b>	This is a reserved Java(TM) programming language keyword. However, it is not used by current versions of the Java programming language.
<b>continue</b>	A Java(TM) programming language keyword used to resume program execution at the end of the current loop. If followed by a label, <b>continue</b> resumes execution where the label occurs.

<b>Default</b>	A Java(TM) programming language keyword optionally used after all <b>case</b> conditions in a <b>switch</b> statement. If all <b>case</b> conditions are not matched by the value of the <b>switch</b> variable, the <b>default</b> keyword will be executed.
<b>do</b>	A Java(TM) programming language keyword used to declare a loop that will iterate a block of statements. The loop's exit condition can be specified with the <b>while</b> keyword.
<b>double</b>	A Java(TM) programming language keyword used to define a variable of type <b>double</b> .
<b>else</b>	A Java(TM) programming language keyword used to execute a block of statements in the case that the test condition with the <b>if</b> keyword evaluates to false.
<b>extends</b>	Class X <b>extends</b> class Y to add functionality, either by adding fields or methods to class Y, or by overriding methods of class Y. An interface <b>extends</b> another <b>interface</b> by adding methods. Class X is said to be a subclass of class Y.
<b>final</b>	A Java(TM) programming language keyword. You define an entity once and cannot change it or derive from it later. More specifically, a <b>final</b> class cannot be subclassed, a <b>final</b> method cannot be overridden and a <b>final</b> variable cannot change from its initialized value.
<b>finally</b>	A Java(TM) programming language keyword that executes a block of statements regardless of whether a Java exception, or runtime error, occurred in a block defined previously by the <b>try</b> keyword.
<b>float</b>	A Java(TM) programming language keyword used to define a floating point number variable.
<b>for</b>	A Java(TM) programming language keyword used to declare a loop that reiterates statements. The programmer can specify the statements to be executed, exit conditions, and initialization variables for the loop.
<b>goto</b>	This is a reserved Java(TM) programming language keyword. However, it is not used by current versions of the Java(TM) programming language.
<b>if</b>	A Java(TM) programming language keyword used to conduct a conditional test and execute a block of statements if the test evaluates to true.
<b>implements</b>	A Java(TM) programming language keyword optionally included in the class declaration to specify any interfaces that are implemented by the current class.
<b>import</b>	A Java(TM) programming language keyword used at the beginning of a source file that can specify classes or entire packages to be referred to later without including their package names in the reference.
<b>instanceof</b>	A two-argument Java(TM) programming language keyword that tests whether the run-time type of its first argument is assignment compatible with its second argument.
<b>int</b>	A Java(TM) programming language keyword used to define a variable of type integer.
<b>interface</b>	A Java(TM) programming language keyword used to define a collection of method definitions and constant values. It can later be implemented by classes that define this interface with the <b>implements</b> keyword.
<b>long</b>	A Java(TM) programming language keyword used to define a variable of type long.

<b>native</b>	A Java(TM) programming language keyword that is used in method declarations to specify that the method is not implemented in the same Java source file, but rather in another language.
<b>new</b>	A Java(TM) programming language keyword used to create an instance of a class.
<b>package</b>	A group of types. Packages are declared with the <b>package</b> keyword.
<b>private</b>	A Java(TM) programming language keyword used in a method or variable declaration. It signifies that the method or variable can only be accessed by other elements of its class.
<b>protected</b>	A Java(TM) programming language keyword used in a method or variable declaration. It signifies that the method or variable can only be accessed by elements residing in its class, subclasses, or classes in the same package.
<b>public</b>	A Java(TM) programming language keyword used in a method or variable declaration. It signifies that the method or variable can be accessed by elements residing in other classes.
<b>return</b>	A Java(TM) programming language keyword used to finish the execution of a method. It can be followed by a value required by the method definition.
<b>short</b>	A Java(TM) programming language keyword used to define a variable of type short.
<b>static</b>	A Java(TM) programming language keyword used to define a variable as a class variable. Classes maintain one copy of class variables regardless of how many instances exist of that class. <b>static</b> can also be used to define a method as a class method. Class methods are invoked by the class instead of a specific instance, and can only operate on class variables.
<b>strictfp</b>	A Java(TM) programming language keyword used to control over floating-point arithmetic.
<b>super</b>	A Java(TM) programming language keyword used to access members of a class inherited by the class in which it appears.
<b>switch</b>	A Java(TM) programming language keyword used to evaluate a variable that can later be matched with a value specified by the <b>case</b> keyword in order to execute a group of statements.
<b>synchronized</b>	A keyword in the Java(TM) programming language that, when applied to a method or code block, guarantees that at most one thread at a time executes that code.
<b>this</b>	A Java(TM) programming language keyword that can be used to represent an instance of the class in which it appears. <b>this</b> can be used to access class variables and methods.
<b>throw</b>	A Java(TM) programming language keyword that allows the user to <b>throw</b> an exception or any class that <b>implements</b> the "throwable" interface.
<b>throws</b>	A Java(TM) programming language keyword used in method declarations that specify which exceptions are not handled within the method but rather passed to the next higher level of the program.

<b>Transient</b>	A keyword in the Java(TM) programming language that indicates that a field is not part of the serialized form of an object. When an object is serialized, the values of its transient fields are not included in the serial representation, while the values of its non-transient fields are included.
<b>try</b>	A Java(TM) programming language keyword that defines a block of statements that may <b>throw</b> a Java language exception. If an exception is thrown, an optional <b>catch</b> block can handle specific exceptions thrown within the <b>try</b> block. Also, an optional <b>finally</b> block will be executed regardless of whether an exception is thrown or not.
<b>void</b>	A Java(TM) programming language keyword used in method declarations to specify that the method does not return any value. <b>void</b> can also be used as a nonfunctional statement.
<b>volatile</b>	A Java(TM) programming language keyword used in variable declarations that specifies that the variable is modified asynchronously by concurrently running threads.
<b>while</b>	A Java(TM) programming language keyword used to declare a loop that iterates a block of statements. The loop's exit condition is specified as part of the while statement.