| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/lang/Thread.html) | **Use** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV   NEXT | [**FRAMES**](http://docs.google.com/index.html?java/lang//class-useThread.html)    [**NO FRAMES**](http://docs.google.com/Thread.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

**Uses of Class**

**java.lang.Thread**

| Packages that use [Thread](http://docs.google.com/java/lang/Thread.html) | |
| --- | --- |
| [**java.lang**](#3znysh7) | Provides classes that are fundamental to the design of the Java programming language. |
| [**java.util.concurrent**](#2et92p0) | Utility classes commonly useful in concurrent programming. |
| [**java.util.concurrent.locks**](#tyjcwt) | Interfaces and classes providing a framework for locking and waiting for conditions that is distinct from built-in synchronization and monitors. |
| [**javax.swing.text**](#3dy6vkm) | Provides classes and interfaces that deal with editable and noneditable text components. |

| Uses of [Thread](http://docs.google.com/java/lang/Thread.html) in [java.lang](http://docs.google.com/java/lang/package-summary.html) | |
| --- | --- |

| Methods in [java.lang](http://docs.google.com/java/lang/package-summary.html) that return [Thread](http://docs.google.com/java/lang/Thread.html) | |
| --- | --- |
| static [Thread](http://docs.google.com/java/lang/Thread.html) | **Thread.**[**currentThread**](http://docs.google.com/java/lang/Thread.html#currentThread())()            Returns a reference to the currently executing thread object. |

| Methods in [java.lang](http://docs.google.com/java/lang/package-summary.html) that return types with arguments of type [Thread](http://docs.google.com/java/lang/Thread.html) | |
| --- | --- |
| static [Map](http://docs.google.com/java/util/Map.html)<[Thread](http://docs.google.com/java/lang/Thread.html),[StackTraceElement](http://docs.google.com/java/lang/StackTraceElement.html)[]> | **Thread.**[**getAllStackTraces**](http://docs.google.com/java/lang/Thread.html#getAllStackTraces())()            Returns a map of stack traces for all live threads. |

| Methods in [java.lang](http://docs.google.com/java/lang/package-summary.html) with parameters of type [Thread](http://docs.google.com/java/lang/Thread.html) | |
| --- | --- |
| void | **Runtime.**[**addShutdownHook**](http://docs.google.com/java/lang/Runtime.html#addShutdownHook(java.lang.Thread))([Thread](http://docs.google.com/java/lang/Thread.html) hook)            Registers a new virtual-machine shutdown hook. |
| void | **SecurityManager.**[**checkAccess**](http://docs.google.com/java/lang/SecurityManager.html#checkAccess(java.lang.Thread))([Thread](http://docs.google.com/java/lang/Thread.html) t)            Throws a SecurityException if the calling thread is not allowed to modify the thread argument. |
| static int | **Thread.**[**enumerate**](http://docs.google.com/java/lang/Thread.html#enumerate(java.lang.Thread%5B%5D))([Thread](http://docs.google.com/java/lang/Thread.html)[] tarray)            Copies into the specified array every active thread in the current thread's thread group and its subgroups. |
| int | **ThreadGroup.**[**enumerate**](http://docs.google.com/java/lang/ThreadGroup.html#enumerate(java.lang.Thread%5B%5D))([Thread](http://docs.google.com/java/lang/Thread.html)[] list)            Copies into the specified array every active thread in this thread group and its subgroups. |
| int | **ThreadGroup.**[**enumerate**](http://docs.google.com/java/lang/ThreadGroup.html#enumerate(java.lang.Thread%5B%5D,%20boolean))([Thread](http://docs.google.com/java/lang/Thread.html)[] list, boolean recurse)            Copies into the specified array every active thread in this thread group. |
| boolean | **Runtime.**[**removeShutdownHook**](http://docs.google.com/java/lang/Runtime.html#removeShutdownHook(java.lang.Thread))([Thread](http://docs.google.com/java/lang/Thread.html) hook)            De-registers a previously-registered virtual-machine shutdown hook. |
| void | **Thread.UncaughtExceptionHandler.**[**uncaughtException**](http://docs.google.com/java/lang/Thread.UncaughtExceptionHandler.html#uncaughtException(java.lang.Thread,%20java.lang.Throwable))([Thread](http://docs.google.com/java/lang/Thread.html) t, [Throwable](http://docs.google.com/java/lang/Throwable.html) e)            Method invoked when the given thread terminates due to the given uncaught exception. |
| void | **ThreadGroup.**[**uncaughtException**](http://docs.google.com/java/lang/ThreadGroup.html#uncaughtException(java.lang.Thread,%20java.lang.Throwable))([Thread](http://docs.google.com/java/lang/Thread.html) t, [Throwable](http://docs.google.com/java/lang/Throwable.html) e)            Called by the Java Virtual Machine when a thread in this thread group stops because of an uncaught exception, and the thread does not have a specific [Thread.UncaughtExceptionHandler](http://docs.google.com/java/lang/Thread.UncaughtExceptionHandler.html) installed. |

| Uses of [Thread](http://docs.google.com/java/lang/Thread.html) in [java.util.concurrent](http://docs.google.com/java/util/concurrent/package-summary.html) | |
| --- | --- |

| Methods in [java.util.concurrent](http://docs.google.com/java/util/concurrent/package-summary.html) that return [Thread](http://docs.google.com/java/lang/Thread.html) | |
| --- | --- |
| [Thread](http://docs.google.com/java/lang/Thread.html) | **ThreadFactory.**[**newThread**](http://docs.google.com/java/util/concurrent/ThreadFactory.html#newThread(java.lang.Runnable))([Runnable](http://docs.google.com/java/lang/Runnable.html) r)            Constructs a new Thread. |

| Methods in [java.util.concurrent](http://docs.google.com/java/util/concurrent/package-summary.html) that return types with arguments of type [Thread](http://docs.google.com/java/lang/Thread.html) | |
| --- | --- |
| protected  [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **Semaphore.**[**getQueuedThreads**](http://docs.google.com/java/util/concurrent/Semaphore.html#getQueuedThreads())()            Returns a collection containing threads that may be waiting to acquire. |

| Methods in [java.util.concurrent](http://docs.google.com/java/util/concurrent/package-summary.html) with parameters of type [Thread](http://docs.google.com/java/lang/Thread.html) | |
| --- | --- |
| protected  void | **ThreadPoolExecutor.**[**beforeExecute**](http://docs.google.com/java/util/concurrent/ThreadPoolExecutor.html#beforeExecute(java.lang.Thread,%20java.lang.Runnable))([Thread](http://docs.google.com/java/lang/Thread.html) t, [Runnable](http://docs.google.com/java/lang/Runnable.html) r)            Method invoked prior to executing the given Runnable in the given thread. |
| void | **TimeUnit.**[**timedJoin**](http://docs.google.com/java/util/concurrent/TimeUnit.html#timedJoin(java.lang.Thread,%20long))([Thread](http://docs.google.com/java/lang/Thread.html) thread, long timeout)            Performs a timed Thread.join using this time unit. |

| Uses of [Thread](http://docs.google.com/java/lang/Thread.html) in [java.util.concurrent.locks](http://docs.google.com/java/util/concurrent/locks/package-summary.html) | |
| --- | --- |

| Methods in [java.util.concurrent.locks](http://docs.google.com/java/util/concurrent/locks/package-summary.html) that return [Thread](http://docs.google.com/java/lang/Thread.html) | |
| --- | --- |
| protected  [Thread](http://docs.google.com/java/lang/Thread.html) | **AbstractOwnableSynchronizer.**[**getExclusiveOwnerThread**](http://docs.google.com/java/util/concurrent/locks/AbstractOwnableSynchronizer.html#getExclusiveOwnerThread())()            Returns the thread last set by setExclusiveOwnerThread, or null if never set. |
| [Thread](http://docs.google.com/java/lang/Thread.html) | **AbstractQueuedLongSynchronizer.**[**getFirstQueuedThread**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedLongSynchronizer.html#getFirstQueuedThread())()            Returns the first (longest-waiting) thread in the queue, or null if no threads are currently queued. |
| [Thread](http://docs.google.com/java/lang/Thread.html) | **AbstractQueuedSynchronizer.**[**getFirstQueuedThread**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedSynchronizer.html#getFirstQueuedThread())()            Returns the first (longest-waiting) thread in the queue, or null if no threads are currently queued. |
| protected  [Thread](http://docs.google.com/java/lang/Thread.html) | **ReentrantReadWriteLock.**[**getOwner**](http://docs.google.com/java/util/concurrent/locks/ReentrantReadWriteLock.html#getOwner())()            Returns the thread that currently owns the write lock, or null if not owned. |
| protected  [Thread](http://docs.google.com/java/lang/Thread.html) | **ReentrantLock.**[**getOwner**](http://docs.google.com/java/util/concurrent/locks/ReentrantLock.html#getOwner())()            Returns the thread that currently owns this lock, or null if not owned. |

| Methods in [java.util.concurrent.locks](http://docs.google.com/java/util/concurrent/locks/package-summary.html) that return types with arguments of type [Thread](http://docs.google.com/java/lang/Thread.html) | |
| --- | --- |
| [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **AbstractQueuedLongSynchronizer.**[**getExclusiveQueuedThreads**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedLongSynchronizer.html#getExclusiveQueuedThreads())()            Returns a collection containing threads that may be waiting to acquire in exclusive mode. |
| [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **AbstractQueuedSynchronizer.**[**getExclusiveQueuedThreads**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedSynchronizer.html#getExclusiveQueuedThreads())()            Returns a collection containing threads that may be waiting to acquire in exclusive mode. |
| protected  [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **ReentrantReadWriteLock.**[**getQueuedReaderThreads**](http://docs.google.com/java/util/concurrent/locks/ReentrantReadWriteLock.html#getQueuedReaderThreads())()            Returns a collection containing threads that may be waiting to acquire the read lock. |
| protected  [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **ReentrantReadWriteLock.**[**getQueuedThreads**](http://docs.google.com/java/util/concurrent/locks/ReentrantReadWriteLock.html#getQueuedThreads())()            Returns a collection containing threads that may be waiting to acquire either the read or write lock. |
| [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **AbstractQueuedLongSynchronizer.**[**getQueuedThreads**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedLongSynchronizer.html#getQueuedThreads())()            Returns a collection containing threads that may be waiting to acquire. |
| [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **AbstractQueuedSynchronizer.**[**getQueuedThreads**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedSynchronizer.html#getQueuedThreads())()            Returns a collection containing threads that may be waiting to acquire. |
| protected  [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **ReentrantLock.**[**getQueuedThreads**](http://docs.google.com/java/util/concurrent/locks/ReentrantLock.html#getQueuedThreads())()            Returns a collection containing threads that may be waiting to acquire this lock. |
| protected  [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **ReentrantReadWriteLock.**[**getQueuedWriterThreads**](http://docs.google.com/java/util/concurrent/locks/ReentrantReadWriteLock.html#getQueuedWriterThreads())()            Returns a collection containing threads that may be waiting to acquire the write lock. |
| [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **AbstractQueuedLongSynchronizer.**[**getSharedQueuedThreads**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedLongSynchronizer.html#getSharedQueuedThreads())()            Returns a collection containing threads that may be waiting to acquire in shared mode. |
| [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **AbstractQueuedSynchronizer.**[**getSharedQueuedThreads**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedSynchronizer.html#getSharedQueuedThreads())()            Returns a collection containing threads that may be waiting to acquire in shared mode. |
| protected  [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **AbstractQueuedLongSynchronizer.ConditionObject.**[**getWaitingThreads**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedLongSynchronizer.ConditionObject.html#getWaitingThreads())()            Returns a collection containing those threads that may be waiting on this Condition. |
| protected  [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **AbstractQueuedSynchronizer.ConditionObject.**[**getWaitingThreads**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedSynchronizer.ConditionObject.html#getWaitingThreads())()            Returns a collection containing those threads that may be waiting on this Condition. |
| [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **AbstractQueuedLongSynchronizer.**[**getWaitingThreads**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedLongSynchronizer.html#getWaitingThreads(java.util.concurrent.locks.AbstractQueuedLongSynchronizer.ConditionObject))([AbstractQueuedLongSynchronizer.ConditionObject](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedLongSynchronizer.ConditionObject.html) condition)            Returns a collection containing those threads that may be waiting on the given condition associated with this synchronizer. |
| [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **AbstractQueuedSynchronizer.**[**getWaitingThreads**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedSynchronizer.html#getWaitingThreads(java.util.concurrent.locks.AbstractQueuedSynchronizer.ConditionObject))([AbstractQueuedSynchronizer.ConditionObject](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedSynchronizer.ConditionObject.html) condition)            Returns a collection containing those threads that may be waiting on the given condition associated with this synchronizer. |
| protected  [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **ReentrantReadWriteLock.**[**getWaitingThreads**](http://docs.google.com/java/util/concurrent/locks/ReentrantReadWriteLock.html#getWaitingThreads(java.util.concurrent.locks.Condition))([Condition](http://docs.google.com/java/util/concurrent/locks/Condition.html) condition)            Returns a collection containing those threads that may be waiting on the given condition associated with the write lock. |
| protected  [Collection](http://docs.google.com/java/util/Collection.html)<[Thread](http://docs.google.com/java/lang/Thread.html)> | **ReentrantLock.**[**getWaitingThreads**](http://docs.google.com/java/util/concurrent/locks/ReentrantLock.html#getWaitingThreads(java.util.concurrent.locks.Condition))([Condition](http://docs.google.com/java/util/concurrent/locks/Condition.html) condition)            Returns a collection containing those threads that may be waiting on the given condition associated with this lock. |

| Methods in [java.util.concurrent.locks](http://docs.google.com/java/util/concurrent/locks/package-summary.html) with parameters of type [Thread](http://docs.google.com/java/lang/Thread.html) | |
| --- | --- |
| static [Object](http://docs.google.com/java/lang/Object.html) | **LockSupport.**[**getBlocker**](http://docs.google.com/java/util/concurrent/locks/LockSupport.html#getBlocker(java.lang.Thread))([Thread](http://docs.google.com/java/lang/Thread.html) t)            Returns the blocker object supplied to the most recent invocation of a park method that has not yet unblocked, or null if not blocked. |
| boolean | **ReentrantReadWriteLock.**[**hasQueuedThread**](http://docs.google.com/java/util/concurrent/locks/ReentrantReadWriteLock.html#hasQueuedThread(java.lang.Thread))([Thread](http://docs.google.com/java/lang/Thread.html) thread)            Queries whether the given thread is waiting to acquire either the read or write lock. |
| boolean | **ReentrantLock.**[**hasQueuedThread**](http://docs.google.com/java/util/concurrent/locks/ReentrantLock.html#hasQueuedThread(java.lang.Thread))([Thread](http://docs.google.com/java/lang/Thread.html) thread)            Queries whether the given thread is waiting to acquire this lock. |
| boolean | **AbstractQueuedLongSynchronizer.**[**isQueued**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedLongSynchronizer.html#isQueued(java.lang.Thread))([Thread](http://docs.google.com/java/lang/Thread.html) thread)            Returns true if the given thread is currently queued. |
| boolean | **AbstractQueuedSynchronizer.**[**isQueued**](http://docs.google.com/java/util/concurrent/locks/AbstractQueuedSynchronizer.html#isQueued(java.lang.Thread))([Thread](http://docs.google.com/java/lang/Thread.html) thread)            Returns true if the given thread is currently queued. |
| protected  void | **AbstractOwnableSynchronizer.**[**setExclusiveOwnerThread**](http://docs.google.com/java/util/concurrent/locks/AbstractOwnableSynchronizer.html#setExclusiveOwnerThread(java.lang.Thread))([Thread](http://docs.google.com/java/lang/Thread.html) t)            Sets the thread that currently owns exclusive access. |
| static void | **LockSupport.**[**unpark**](http://docs.google.com/java/util/concurrent/locks/LockSupport.html#unpark(java.lang.Thread))([Thread](http://docs.google.com/java/lang/Thread.html) thread)            Makes available the permit for the given thread, if it was not already available. |

| Uses of [Thread](http://docs.google.com/java/lang/Thread.html) in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) | |
| --- | --- |

| Methods in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) that return [Thread](http://docs.google.com/java/lang/Thread.html) | |
| --- | --- |
| protected  [Thread](http://docs.google.com/java/lang/Thread.html) | **AbstractDocument.**[**getCurrentWriter**](http://docs.google.com/javax/swing/text/AbstractDocument.html#getCurrentWriter())()            Fetches the current writing thread if there is one. |

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/lang/Thread.html) | **Use** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV   NEXT | [**FRAMES**](http://docs.google.com/index.html?java/lang//class-useThread.html)    [**NO FRAMES**](http://docs.google.com/Thread.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).