| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/util/concurrent/BlockingQueue.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/util/concurrent//class-useBlockingQueue.html)    [**NO FRAMES**](http://docs.google.com/BlockingQueue.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

**Uses of Interface**

**java.util.concurrent.BlockingQueue**

| Packages that use [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html) | |
| --- | --- |
| [**java.util.concurrent**](#3znysh7) | Utility classes commonly useful in concurrent programming. |

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

| Subinterfaces of [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html) in [java.util.concurrent](http://docs.google.com/java/util/concurrent/package-summary.html) | |
| --- | --- |
| interface | [**BlockingDeque<E>**](http://docs.google.com/java/util/concurrent/BlockingDeque.html)            A [Deque](http://docs.google.com/java/util/Deque.html) that additionally supports blocking operations that wait for the deque to become non-empty when retrieving an element, and wait for space to become available in the deque when storing an element. |

| Classes in [java.util.concurrent](http://docs.google.com/java/util/concurrent/package-summary.html) that implement [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html) | |
| --- | --- |
| class | [**ArrayBlockingQueue<E>**](http://docs.google.com/java/util/concurrent/ArrayBlockingQueue.html)            A bounded [blocking queue](http://docs.google.com/java/util/concurrent/BlockingQueue.html) backed by an array. |
| class | [**DelayQueue<E extends Delayed>**](http://docs.google.com/java/util/concurrent/DelayQueue.html)            An unbounded [blocking queue](http://docs.google.com/java/util/concurrent/BlockingQueue.html) of Delayed elements, in which an element can only be taken when its delay has expired. |
| class | [**LinkedBlockingDeque<E>**](http://docs.google.com/java/util/concurrent/LinkedBlockingDeque.html)            An optionally-bounded [blocking deque](http://docs.google.com/java/util/concurrent/BlockingDeque.html) based on linked nodes. |
| class | [**LinkedBlockingQueue<E>**](http://docs.google.com/java/util/concurrent/LinkedBlockingQueue.html)            An optionally-bounded [blocking queue](http://docs.google.com/java/util/concurrent/BlockingQueue.html) based on linked nodes. |
| class | [**PriorityBlockingQueue<E>**](http://docs.google.com/java/util/concurrent/PriorityBlockingQueue.html)            An unbounded [blocking queue](http://docs.google.com/java/util/concurrent/BlockingQueue.html) that uses the same ordering rules as class [PriorityQueue](http://docs.google.com/java/util/PriorityQueue.html) and supplies blocking retrieval operations. |
| class | [**SynchronousQueue<E>**](http://docs.google.com/java/util/concurrent/SynchronousQueue.html)            A [blocking queue](http://docs.google.com/java/util/concurrent/BlockingQueue.html) in which each insert operation must wait for a corresponding remove operation by another thread, and vice versa. |

| Methods in [java.util.concurrent](http://docs.google.com/java/util/concurrent/package-summary.html) that return [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html) | |
| --- | --- |
| [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html)<[Runnable](http://docs.google.com/java/lang/Runnable.html)> | **ScheduledThreadPoolExecutor.**[**getQueue**](http://docs.google.com/java/util/concurrent/ScheduledThreadPoolExecutor.html#getQueue())()            Returns the task queue used by this executor. |
| [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html)<[Runnable](http://docs.google.com/java/lang/Runnable.html)> | **ThreadPoolExecutor.**[**getQueue**](http://docs.google.com/java/util/concurrent/ThreadPoolExecutor.html#getQueue())()            Returns the task queue used by this executor. |

| Constructors in [java.util.concurrent](http://docs.google.com/java/util/concurrent/package-summary.html) with parameters of type [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html) | |
| --- | --- |
| [**ExecutorCompletionService**](http://docs.google.com/java/util/concurrent/ExecutorCompletionService.html#ExecutorCompletionService(java.util.concurrent.Executor,%20java.util.concurrent.BlockingQueue))([Executor](http://docs.google.com/java/util/concurrent/Executor.html) executor, [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html)<[Future](http://docs.google.com/java/util/concurrent/Future.html)<[V](http://docs.google.com/java/util/concurrent/ExecutorCompletionService.html)>> completionQueue)            Creates an ExecutorCompletionService using the supplied executor for base task execution and the supplied queue as its completion queue. |
| [**ThreadPoolExecutor**](http://docs.google.com/java/util/concurrent/ThreadPoolExecutor.html#ThreadPoolExecutor(int,%20int,%20long,%20java.util.concurrent.TimeUnit,%20java.util.concurrent.BlockingQueue))(int corePoolSize, int maximumPoolSize, long keepAliveTime, [TimeUnit](http://docs.google.com/java/util/concurrent/TimeUnit.html) unit, [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html)<[Runnable](http://docs.google.com/java/lang/Runnable.html)> workQueue)            Creates a new ThreadPoolExecutor with the given initial parameters and default thread factory and rejected execution handler. |
| [**ThreadPoolExecutor**](http://docs.google.com/java/util/concurrent/ThreadPoolExecutor.html#ThreadPoolExecutor(int,%20int,%20long,%20java.util.concurrent.TimeUnit,%20java.util.concurrent.BlockingQueue,%20java.util.concurrent.RejectedExecutionHandler))(int corePoolSize, int maximumPoolSize, long keepAliveTime, [TimeUnit](http://docs.google.com/java/util/concurrent/TimeUnit.html) unit, [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html)<[Runnable](http://docs.google.com/java/lang/Runnable.html)> workQueue, [RejectedExecutionHandler](http://docs.google.com/java/util/concurrent/RejectedExecutionHandler.html) handler)            Creates a new ThreadPoolExecutor with the given initial parameters and default thread factory. |
| [**ThreadPoolExecutor**](http://docs.google.com/java/util/concurrent/ThreadPoolExecutor.html#ThreadPoolExecutor(int,%20int,%20long,%20java.util.concurrent.TimeUnit,%20java.util.concurrent.BlockingQueue,%20java.util.concurrent.ThreadFactory))(int corePoolSize, int maximumPoolSize, long keepAliveTime, [TimeUnit](http://docs.google.com/java/util/concurrent/TimeUnit.html) unit, [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html)<[Runnable](http://docs.google.com/java/lang/Runnable.html)> workQueue, [ThreadFactory](http://docs.google.com/java/util/concurrent/ThreadFactory.html) threadFactory)            Creates a new ThreadPoolExecutor with the given initial parameters and default rejected execution handler. |
| [**ThreadPoolExecutor**](http://docs.google.com/java/util/concurrent/ThreadPoolExecutor.html#ThreadPoolExecutor(int,%20int,%20long,%20java.util.concurrent.TimeUnit,%20java.util.concurrent.BlockingQueue,%20java.util.concurrent.ThreadFactory,%20java.util.concurrent.RejectedExecutionHandler))(int corePoolSize, int maximumPoolSize, long keepAliveTime, [TimeUnit](http://docs.google.com/java/util/concurrent/TimeUnit.html) unit, [BlockingQueue](http://docs.google.com/java/util/concurrent/BlockingQueue.html)<[Runnable](http://docs.google.com/java/lang/Runnable.html)> workQueue, [ThreadFactory](http://docs.google.com/java/util/concurrent/ThreadFactory.html) threadFactory, [RejectedExecutionHandler](http://docs.google.com/java/util/concurrent/RejectedExecutionHandler.html) handler)            Creates a new ThreadPoolExecutor with the given initial parameters. |

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/util/concurrent/BlockingQueue.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/util/concurrent//class-useBlockingQueue.html)    [**NO FRAMES**](http://docs.google.com/BlockingQueue.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).