| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Thread.State.html) | [**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 CLASS**](http://docs.google.com/java/lang/Thread.html)   [**NEXT CLASS**](http://docs.google.com/java/lang/Thread.UncaughtExceptionHandler.html) | [**FRAMES**](http://docs.google.com/index.html?java/lang/Thread.State.html)    [**NO FRAMES**](http://docs.google.com/Thread.State.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [ENUM CONSTANTS](#3znysh7) | FIELD | [METHOD](#2et92p0) | DETAIL: [ENUM CONSTANTS](#1t3h5sf) | FIELD | [METHOD](#35nkun2) |

## **java.lang**

Enum Thread.State

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.lang.Enum](http://docs.google.com/java/lang/Enum.html)<[Thread.State](http://docs.google.com/java/lang/Thread.State.html)>  
 **java.lang.Thread.State**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [Comparable](http://docs.google.com/java/lang/Comparable.html)<[Thread.State](http://docs.google.com/java/lang/Thread.State.html)> **Enclosing class:**[Thread](http://docs.google.com/java/lang/Thread.html)

public static enum **Thread.State**extends [Enum](http://docs.google.com/java/lang/Enum.html)<[Thread.State](http://docs.google.com/java/lang/Thread.State.html)>

A thread state. A thread can be in one of the following states:

* [NEW](http://docs.google.com/java/lang/Thread.State.html#NEW)  
  A thread that has not yet started is in this state.
* [RUNNABLE](http://docs.google.com/java/lang/Thread.State.html#RUNNABLE)  
  A thread executing in the Java virtual machine is in this state.
* [BLOCKED](http://docs.google.com/java/lang/Thread.State.html#BLOCKED)  
  A thread that is blocked waiting for a monitor lock is in this state.
* [WAITING](http://docs.google.com/java/lang/Thread.State.html#WAITING)  
  A thread that is waiting indefinitely for another thread to perform a particular action is in this state.
* [TIMED\_WAITING](http://docs.google.com/java/lang/Thread.State.html#TIMED_WAITING)  
  A thread that is waiting for another thread to perform an action for up to a specified waiting time is in this state.
* [TERMINATED](http://docs.google.com/java/lang/Thread.State.html#TERMINATED)  
  A thread that has exited is in this state.

A thread can be in only one state at a given point in time. These states are virtual machine states which do not reflect any operating system thread states.

**Since:** 1.5 **See Also:**[Thread.getState()](http://docs.google.com/java/lang/Thread.html#getState())

| **Enum Constant Summary** | |
| --- | --- |
| [**BLOCKED**](http://docs.google.com/java/lang/Thread.State.html#BLOCKED)            Thread state for a thread blocked waiting for a monitor lock. |
| [**NEW**](http://docs.google.com/java/lang/Thread.State.html#NEW)            Thread state for a thread which has not yet started. |
| [**RUNNABLE**](http://docs.google.com/java/lang/Thread.State.html#RUNNABLE)            Thread state for a runnable thread. |
| [**TERMINATED**](http://docs.google.com/java/lang/Thread.State.html#TERMINATED)            Thread state for a terminated thread. |
| [**TIMED\_WAITING**](http://docs.google.com/java/lang/Thread.State.html#TIMED_WAITING)            Thread state for a waiting thread with a specified waiting time. |
| [**WAITING**](http://docs.google.com/java/lang/Thread.State.html#WAITING)            Thread state for a waiting thread. |

| **Method Summary** | |
| --- | --- |
| static [Thread.State](http://docs.google.com/java/lang/Thread.State.html) | [**valueOf**](http://docs.google.com/java/lang/Thread.State.html#valueOf(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Returns the enum constant of this type with the specified name. |
| static [Thread.State](http://docs.google.com/java/lang/Thread.State.html)[] | [**values**](http://docs.google.com/java/lang/Thread.State.html#values())()            Returns an array containing the constants of this enum type, in the order they are declared. |

| **Methods inherited from class java.lang.**[**Enum**](http://docs.google.com/java/lang/Enum.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Enum.html#clone()), [compareTo](http://docs.google.com/java/lang/Enum.html#compareTo(E)), [equals](http://docs.google.com/java/lang/Enum.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Enum.html#finalize()), [getDeclaringClass](http://docs.google.com/java/lang/Enum.html#getDeclaringClass()), [hashCode](http://docs.google.com/java/lang/Enum.html#hashCode()), [name](http://docs.google.com/java/lang/Enum.html#name()), [ordinal](http://docs.google.com/java/lang/Enum.html#ordinal()), [toString](http://docs.google.com/java/lang/Enum.html#toString()), [valueOf](http://docs.google.com/java/lang/Enum.html#valueOf(java.lang.Class,%20java.lang.String)) |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Enum Constant Detail** |
| --- |

### NEW

public static final [Thread.State](http://docs.google.com/java/lang/Thread.State.html) **NEW**

Thread state for a thread which has not yet started.

### RUNNABLE

public static final [Thread.State](http://docs.google.com/java/lang/Thread.State.html) **RUNNABLE**

Thread state for a runnable thread. A thread in the runnable state is executing in the Java virtual machine but it may be waiting for other resources from the operating system such as processor.

### BLOCKED

public static final [Thread.State](http://docs.google.com/java/lang/Thread.State.html) **BLOCKED**

Thread state for a thread blocked waiting for a monitor lock. A thread in the blocked state is waiting for a monitor lock to enter a synchronized block/method or reenter a synchronized block/method after calling [Object.wait](http://docs.google.com/java/lang/Object.html#wait()).

### WAITING

public static final [Thread.State](http://docs.google.com/java/lang/Thread.State.html) **WAITING**

Thread state for a waiting thread. A thread is in the waiting state due to calling one of the following methods:

* [Object.wait](http://docs.google.com/java/lang/Object.html#wait()) with no timeout
* [Thread.join](http://docs.google.com/java/lang/Thread.html#join()) with no timeout
* [LockSupport.park](http://docs.google.com/java/util/concurrent/locks/LockSupport.html#park())

A thread in the waiting state is waiting for another thread to perform a particular action. For example, a thread that has called Object.wait() on an object is waiting for another thread to call Object.notify() or Object.notifyAll() on that object. A thread that has called Thread.join() is waiting for a specified thread to terminate.

### TIMED\_WAITING

public static final [Thread.State](http://docs.google.com/java/lang/Thread.State.html) **TIMED\_WAITING**

Thread state for a waiting thread with a specified waiting time. A thread is in the timed waiting state due to calling one of the following methods with a specified positive waiting time:

* [Thread.sleep](http://docs.google.com/java/lang/Thread.html#sleep(long))
* [Object.wait](http://docs.google.com/java/lang/Object.html#wait(long)) with timeout
* [Thread.join](http://docs.google.com/java/lang/Thread.html#join(long)) with timeout
* [LockSupport.parkNanos](http://docs.google.com/java/util/concurrent/locks/LockSupport.html#parkNanos(java.lang.Object,%20long))
* [LockSupport.parkUntil](http://docs.google.com/java/util/concurrent/locks/LockSupport.html#parkUntil(java.lang.Object,%20long))

### TERMINATED

public static final [Thread.State](http://docs.google.com/java/lang/Thread.State.html) **TERMINATED**

Thread state for a terminated thread. The thread has completed execution.

| **Method Detail** |
| --- |

### values

public static [Thread.State](http://docs.google.com/java/lang/Thread.State.html)[] **values**()

Returns an array containing the constants of this enum type, in the order they are declared. This method may be used to iterate over the constants as follows:

for (Thread.State c : Thread.State.values())  
  System.out.println(c);

**Returns:**an array containing the constants of this enum type, in the order they are declared

### valueOf

public static [Thread.State](http://docs.google.com/java/lang/Thread.State.html) **valueOf**([String](http://docs.google.com/java/lang/String.html) name)

Returns the enum constant of this type with the specified name. The string must match *exactly* an identifier used to declare an enum constant in this type. (Extraneous whitespace characters are not permitted.)

**Parameters:**name - the name of the enum constant to be returned. **Returns:**the enum constant with the specified name **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if this enum type has no constant with the specified name [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the argument is null

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Thread.State.html) | [**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 CLASS**](http://docs.google.com/java/lang/Thread.html)   [**NEXT CLASS**](http://docs.google.com/java/lang/Thread.UncaughtExceptionHandler.html) | [**FRAMES**](http://docs.google.com/index.html?java/lang/Thread.State.html)    [**NO FRAMES**](http://docs.google.com/Thread.State.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [ENUM CONSTANTS](#3znysh7) | FIELD | [METHOD](#2et92p0) | DETAIL: [ENUM CONSTANTS](#1t3h5sf) | FIELD | [METHOD](#35nkun2) |

[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).