| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/TimerMBean.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/javax/management/timer/Timer.html)   [**NEXT CLASS**](http://docs.google.com/javax/management/timer/TimerNotification.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/timer/TimerMBean.html)    [**NO FRAMES**](http://docs.google.com/TimerMBean.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **javax.management.timer**

Interface TimerMBean

**All Known Implementing Classes:** [Timer](http://docs.google.com/javax/management/timer/Timer.html)

public interface **TimerMBean**

Exposes the management interface of the timer MBean.

**Since:** 1.5

| **Method Summary** | |
| --- | --- |
| [Integer](http://docs.google.com/java/lang/Integer.html) | [**addNotification**](http://docs.google.com/javax/management/timer/TimerMBean.html#addNotification(java.lang.String,%20java.lang.String,%20java.lang.Object,%20java.util.Date))([String](http://docs.google.com/java/lang/String.html) type, [String](http://docs.google.com/java/lang/String.html) message, [Object](http://docs.google.com/java/lang/Object.html) userData, [Date](http://docs.google.com/java/util/Date.html) date)            Creates a new timer notification with the specified type, message and userData and inserts it into the list of notifications with a given date and a null period and number of occurrences. |
| [Integer](http://docs.google.com/java/lang/Integer.html) | [**addNotification**](http://docs.google.com/javax/management/timer/TimerMBean.html#addNotification(java.lang.String,%20java.lang.String,%20java.lang.Object,%20java.util.Date,%20long))([String](http://docs.google.com/java/lang/String.html) type, [String](http://docs.google.com/java/lang/String.html) message, [Object](http://docs.google.com/java/lang/Object.html) userData, [Date](http://docs.google.com/java/util/Date.html) date, long period)            Creates a new timer notification with the specified type, message and userData and inserts it into the list of notifications with a given date and period and a null number of occurrences. |
| [Integer](http://docs.google.com/java/lang/Integer.html) | [**addNotification**](http://docs.google.com/javax/management/timer/TimerMBean.html#addNotification(java.lang.String,%20java.lang.String,%20java.lang.Object,%20java.util.Date,%20long,%20long))([String](http://docs.google.com/java/lang/String.html) type, [String](http://docs.google.com/java/lang/String.html) message, [Object](http://docs.google.com/java/lang/Object.html) userData, [Date](http://docs.google.com/java/util/Date.html) date, long period, long nbOccurences)            Creates a new timer notification with the specified type, message and userData and inserts it into the list of notifications with a given date, period and number of occurrences. |
| [Integer](http://docs.google.com/java/lang/Integer.html) | [**addNotification**](http://docs.google.com/javax/management/timer/TimerMBean.html#addNotification(java.lang.String,%20java.lang.String,%20java.lang.Object,%20java.util.Date,%20long,%20long,%20boolean))([String](http://docs.google.com/java/lang/String.html) type, [String](http://docs.google.com/java/lang/String.html) message, [Object](http://docs.google.com/java/lang/Object.html) userData, [Date](http://docs.google.com/java/util/Date.html) date, long period, long nbOccurences, boolean fixedRate)            Creates a new timer notification with the specified type, message and userData and inserts it into the list of notifications with a given date, period and number of occurrences. |
| [Vector](http://docs.google.com/java/util/Vector.html)<[Integer](http://docs.google.com/java/lang/Integer.html)> | [**getAllNotificationIDs**](http://docs.google.com/javax/management/timer/TimerMBean.html#getAllNotificationIDs())()            Gets all timer notification identifiers registered into the list of notifications. |
| [Date](http://docs.google.com/java/util/Date.html) | [**getDate**](http://docs.google.com/javax/management/timer/TimerMBean.html#getDate(java.lang.Integer))([Integer](http://docs.google.com/java/lang/Integer.html) id)            Gets a copy of the date associated to a timer notification. |
| [Boolean](http://docs.google.com/java/lang/Boolean.html) | [**getFixedRate**](http://docs.google.com/javax/management/timer/TimerMBean.html#getFixedRate(java.lang.Integer))([Integer](http://docs.google.com/java/lang/Integer.html) id)            Gets a copy of the flag indicating whether a periodic notification is executed at *fixed-delay* or at *fixed-rate*. |
| int | [**getNbNotifications**](http://docs.google.com/javax/management/timer/TimerMBean.html#getNbNotifications())()            Gets the number of timer notifications registered into the list of notifications. |
| [Long](http://docs.google.com/java/lang/Long.html) | [**getNbOccurences**](http://docs.google.com/javax/management/timer/TimerMBean.html#getNbOccurences(java.lang.Integer))([Integer](http://docs.google.com/java/lang/Integer.html) id)            Gets a copy of the remaining number of occurrences associated to a timer notification. |
| [Vector](http://docs.google.com/java/util/Vector.html)<[Integer](http://docs.google.com/java/lang/Integer.html)> | [**getNotificationIDs**](http://docs.google.com/javax/management/timer/TimerMBean.html#getNotificationIDs(java.lang.String))([String](http://docs.google.com/java/lang/String.html) type)            Gets all the identifiers of timer notifications corresponding to the specified type. |
| [String](http://docs.google.com/java/lang/String.html) | [**getNotificationMessage**](http://docs.google.com/javax/management/timer/TimerMBean.html#getNotificationMessage(java.lang.Integer))([Integer](http://docs.google.com/java/lang/Integer.html) id)            Gets the timer notification detailed message corresponding to the specified identifier. |
| [String](http://docs.google.com/java/lang/String.html) | [**getNotificationType**](http://docs.google.com/javax/management/timer/TimerMBean.html#getNotificationType(java.lang.Integer))([Integer](http://docs.google.com/java/lang/Integer.html) id)            Gets the timer notification type corresponding to the specified identifier. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getNotificationUserData**](http://docs.google.com/javax/management/timer/TimerMBean.html#getNotificationUserData(java.lang.Integer))([Integer](http://docs.google.com/java/lang/Integer.html) id)            Gets the timer notification user data object corresponding to the specified identifier. |
| [Long](http://docs.google.com/java/lang/Long.html) | [**getPeriod**](http://docs.google.com/javax/management/timer/TimerMBean.html#getPeriod(java.lang.Integer))([Integer](http://docs.google.com/java/lang/Integer.html) id)            Gets a copy of the period (in milliseconds) associated to a timer notification. |
| boolean | [**getSendPastNotifications**](http://docs.google.com/javax/management/timer/TimerMBean.html#getSendPastNotifications())()            Gets the flag indicating whether or not the timer sends past notifications. |
| boolean | [**isActive**](http://docs.google.com/javax/management/timer/TimerMBean.html#isActive())()            Tests whether the timer MBean is active. |
| boolean | [**isEmpty**](http://docs.google.com/javax/management/timer/TimerMBean.html#isEmpty())()            Tests whether the list of timer notifications is empty. |
| void | [**removeAllNotifications**](http://docs.google.com/javax/management/timer/TimerMBean.html#removeAllNotifications())()            Removes all the timer notifications from the list of notifications and resets the counter used to update the timer notification identifiers. |
| void | [**removeNotification**](http://docs.google.com/javax/management/timer/TimerMBean.html#removeNotification(java.lang.Integer))([Integer](http://docs.google.com/java/lang/Integer.html) id)            Removes the timer notification corresponding to the specified identifier from the list of notifications. |
| void | [**removeNotifications**](http://docs.google.com/javax/management/timer/TimerMBean.html#removeNotifications(java.lang.String))([String](http://docs.google.com/java/lang/String.html) type)            Removes all the timer notifications corresponding to the specified type from the list of notifications. |
| void | [**setSendPastNotifications**](http://docs.google.com/javax/management/timer/TimerMBean.html#setSendPastNotifications(boolean))(boolean value)            Sets the flag indicating whether the timer sends past notifications or not. |
| void | [**start**](http://docs.google.com/javax/management/timer/TimerMBean.html#start())()            Starts the timer. |
| void | [**stop**](http://docs.google.com/javax/management/timer/TimerMBean.html#stop())()            Stops the timer. |

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

### start

void **start**()

Starts the timer.

If there is one or more timer notifications before the time in the list of notifications, the notification is sent according to the sendPastNotifications flag and then, updated according to its period and remaining number of occurrences. If the timer notification date remains earlier than the current date, this notification is just removed from the list of notifications.

### stop

void **stop**()

Stops the timer.

### addNotification

[Integer](http://docs.google.com/java/lang/Integer.html) **addNotification**([String](http://docs.google.com/java/lang/String.html) type,  
 [String](http://docs.google.com/java/lang/String.html) message,  
 [Object](http://docs.google.com/java/lang/Object.html) userData,  
 [Date](http://docs.google.com/java/util/Date.html) date,  
 long period,  
 long nbOccurences,  
 boolean fixedRate)  
 throws [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html)

Creates a new timer notification with the specified type, message and userData and inserts it into the list of notifications with a given date, period and number of occurrences.

If the timer notification to be inserted has a date that is before the current date, the method behaves as if the specified date were the current date.

For once-off notifications, the notification is delivered immediately.

For periodic notifications, the first notification is delivered immediately and the subsequent ones are spaced as specified by the period parameter.

Note that once the timer notification has been added into the list of notifications, its associated date, period and number of occurrences cannot be updated.

In the case of a periodic notification, the value of parameter *fixedRate* is used to specify the execution scheme, as specified in [Timer](http://docs.google.com/java/util/Timer.html).

**Parameters:**type - The timer notification type.message - The timer notification detailed message.userData - The timer notification user data object.date - The date when the notification occurs.period - The period of the timer notification (in milliseconds).nbOccurences - The total number the timer notification will be emitted.fixedRate - If true and if the notification is periodic, the notification is scheduled with a *fixed-rate* execution scheme. If false and if the notification is periodic, the notification is scheduled with a *fixed-delay* execution scheme. Ignored if the notification is not periodic. **Returns:**The identifier of the new created timer notification. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - The date is null or the period or the number of occurrences is negative.**See Also:**[addNotification(String, String, Object, Date, long, long)](http://docs.google.com/javax/management/timer/TimerMBean.html#addNotification(java.lang.String,%20java.lang.String,%20java.lang.Object,%20java.util.Date,%20long,%20long))

### addNotification

[Integer](http://docs.google.com/java/lang/Integer.html) **addNotification**([String](http://docs.google.com/java/lang/String.html) type,  
 [String](http://docs.google.com/java/lang/String.html) message,  
 [Object](http://docs.google.com/java/lang/Object.html) userData,  
 [Date](http://docs.google.com/java/util/Date.html) date,  
 long period,  
 long nbOccurences)  
 throws [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html)

Creates a new timer notification with the specified type, message and userData and inserts it into the list of notifications with a given date, period and number of occurrences.

If the timer notification to be inserted has a date that is before the current date, the method behaves as if the specified date were the current date.

For once-off notifications, the notification is delivered immediately.

For periodic notifications, the first notification is delivered immediately and the subsequent ones are spaced as specified by the period parameter.

Note that once the timer notification has been added into the list of notifications, its associated date, period and number of occurrences cannot be updated.

In the case of a periodic notification, uses a *fixed-delay* execution scheme, as specified in [Timer](http://docs.google.com/java/util/Timer.html). In order to use a *fixed-rate* execution scheme, use [addNotification(String, String, Object, Date, long, long, boolean)](http://docs.google.com/javax/management/timer/TimerMBean.html#addNotification(java.lang.String,%20java.lang.String,%20java.lang.Object,%20java.util.Date,%20long,%20long,%20boolean)) instead.

**Parameters:**type - The timer notification type.message - The timer notification detailed message.userData - The timer notification user data object.date - The date when the notification occurs.period - The period of the timer notification (in milliseconds).nbOccurences - The total number the timer notification will be emitted. **Returns:**The identifier of the new created timer notification. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - The date is null or the period or the number of occurrences is negative.**See Also:**[addNotification(String, String, Object, Date, long, long, boolean)](http://docs.google.com/javax/management/timer/TimerMBean.html#addNotification(java.lang.String,%20java.lang.String,%20java.lang.Object,%20java.util.Date,%20long,%20long,%20boolean))

### addNotification

[Integer](http://docs.google.com/java/lang/Integer.html) **addNotification**([String](http://docs.google.com/java/lang/String.html) type,  
 [String](http://docs.google.com/java/lang/String.html) message,  
 [Object](http://docs.google.com/java/lang/Object.html) userData,  
 [Date](http://docs.google.com/java/util/Date.html) date,  
 long period)  
 throws [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html)

Creates a new timer notification with the specified type, message and userData and inserts it into the list of notifications with a given date and period and a null number of occurrences.

The timer notification will repeat continuously using the timer period using a *fixed-delay* execution scheme, as specified in [Timer](http://docs.google.com/java/util/Timer.html). In order to use a *fixed-rate* execution scheme, use [addNotification(String, String, Object, Date, long, long, boolean)](http://docs.google.com/javax/management/timer/TimerMBean.html#addNotification(java.lang.String,%20java.lang.String,%20java.lang.Object,%20java.util.Date,%20long,%20long,%20boolean)) instead.

If the timer notification to be inserted has a date that is before the current date, the method behaves as if the specified date were the current date. The first notification is delivered immediately and the subsequent ones are spaced as specified by the period parameter.

**Parameters:**type - The timer notification type.message - The timer notification detailed message.userData - The timer notification user data object.date - The date when the notification occurs.period - The period of the timer notification (in milliseconds). **Returns:**The identifier of the new created timer notification. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - The date is null or the period is negative.

### addNotification

[Integer](http://docs.google.com/java/lang/Integer.html) **addNotification**([String](http://docs.google.com/java/lang/String.html) type,  
 [String](http://docs.google.com/java/lang/String.html) message,  
 [Object](http://docs.google.com/java/lang/Object.html) userData,  
 [Date](http://docs.google.com/java/util/Date.html) date)  
 throws [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html)

Creates a new timer notification with the specified type, message and userData and inserts it into the list of notifications with a given date and a null period and number of occurrences.

The timer notification will be handled once at the specified date.

If the timer notification to be inserted has a date that is before the current date, the method behaves as if the specified date were the current date and the notification is delivered immediately.

**Parameters:**type - The timer notification type.message - The timer notification detailed message.userData - The timer notification user data object.date - The date when the notification occurs. **Returns:**The identifier of the new created timer notification. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - The date is null.

### removeNotification

void **removeNotification**([Integer](http://docs.google.com/java/lang/Integer.html) id)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html)

Removes the timer notification corresponding to the specified identifier from the list of notifications.

**Parameters:**id - The timer notification identifier. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The specified identifier does not correspond to any timer notification in the list of notifications of this timer MBean.

### removeNotifications

void **removeNotifications**([String](http://docs.google.com/java/lang/String.html) type)  
 throws [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html)

Removes all the timer notifications corresponding to the specified type from the list of notifications.

**Parameters:**type - The timer notification type. **Throws:** [InstanceNotFoundException](http://docs.google.com/javax/management/InstanceNotFoundException.html) - The specified type does not correspond to any timer notification in the list of notifications of this timer MBean.

### removeAllNotifications

void **removeAllNotifications**()

Removes all the timer notifications from the list of notifications and resets the counter used to update the timer notification identifiers.

### getNbNotifications

int **getNbNotifications**()

Gets the number of timer notifications registered into the list of notifications.

**Returns:**The number of timer notifications.

### getAllNotificationIDs

[Vector](http://docs.google.com/java/util/Vector.html)<[Integer](http://docs.google.com/java/lang/Integer.html)> **getAllNotificationIDs**()

Gets all timer notification identifiers registered into the list of notifications.

**Returns:**A vector of Integer objects containing all the timer notification identifiers.

The vector is empty if there is no timer notification registered for this timer MBean.

### getNotificationIDs

[Vector](http://docs.google.com/java/util/Vector.html)<[Integer](http://docs.google.com/java/lang/Integer.html)> **getNotificationIDs**([String](http://docs.google.com/java/lang/String.html) type)

Gets all the identifiers of timer notifications corresponding to the specified type.

**Parameters:**type - The timer notification type. **Returns:**A vector of Integer objects containing all the identifiers of timer notifications with the specified type.

The vector is empty if there is no timer notifications registered for this timer MBean with the specified type.

### getNotificationType

[String](http://docs.google.com/java/lang/String.html) **getNotificationType**([Integer](http://docs.google.com/java/lang/Integer.html) id)

Gets the timer notification type corresponding to the specified identifier.

**Parameters:**id - The timer notification identifier. **Returns:**The timer notification type or null if the identifier is not mapped to any timer notification registered for this timer MBean.

### getNotificationMessage

[String](http://docs.google.com/java/lang/String.html) **getNotificationMessage**([Integer](http://docs.google.com/java/lang/Integer.html) id)

Gets the timer notification detailed message corresponding to the specified identifier.

**Parameters:**id - The timer notification identifier. **Returns:**The timer notification detailed message or null if the identifier is not mapped to any timer notification registered for this timer MBean.

### getNotificationUserData

[Object](http://docs.google.com/java/lang/Object.html) **getNotificationUserData**([Integer](http://docs.google.com/java/lang/Integer.html) id)

Gets the timer notification user data object corresponding to the specified identifier.

**Parameters:**id - The timer notification identifier. **Returns:**The timer notification user data object or null if the identifier is not mapped to any timer notification registered for this timer MBean.

### getDate

[Date](http://docs.google.com/java/util/Date.html) **getDate**([Integer](http://docs.google.com/java/lang/Integer.html) id)

Gets a copy of the date associated to a timer notification.

**Parameters:**id - The timer notification identifier. **Returns:**A copy of the date or null if the identifier is not mapped to any timer notification registered for this timer MBean.

### getPeriod

[Long](http://docs.google.com/java/lang/Long.html) **getPeriod**([Integer](http://docs.google.com/java/lang/Integer.html) id)

Gets a copy of the period (in milliseconds) associated to a timer notification.

**Parameters:**id - The timer notification identifier. **Returns:**A copy of the period or null if the identifier is not mapped to any timer notification registered for this timer MBean.

### getNbOccurences

[Long](http://docs.google.com/java/lang/Long.html) **getNbOccurences**([Integer](http://docs.google.com/java/lang/Integer.html) id)

Gets a copy of the remaining number of occurrences associated to a timer notification.

**Parameters:**id - The timer notification identifier. **Returns:**A copy of the remaining number of occurrences or null if the identifier is not mapped to any timer notification registered for this timer MBean.

### getFixedRate

[Boolean](http://docs.google.com/java/lang/Boolean.html) **getFixedRate**([Integer](http://docs.google.com/java/lang/Integer.html) id)

Gets a copy of the flag indicating whether a periodic notification is executed at *fixed-delay* or at *fixed-rate*.

**Parameters:**id - The timer notification identifier. **Returns:**A copy of the flag indicating whether a periodic notification is executed at *fixed-delay* or at *fixed-rate*.

### getSendPastNotifications

boolean **getSendPastNotifications**()

Gets the flag indicating whether or not the timer sends past notifications.

**Returns:**The past notifications sending on/off flag value.**See Also:**[setSendPastNotifications(boolean)](http://docs.google.com/javax/management/timer/TimerMBean.html#setSendPastNotifications(boolean))

### setSendPastNotifications

void **setSendPastNotifications**(boolean value)

Sets the flag indicating whether the timer sends past notifications or not.

**Parameters:**value - The past notifications sending on/off flag value.**See Also:**[getSendPastNotifications()](http://docs.google.com/javax/management/timer/TimerMBean.html#getSendPastNotifications())

### isActive

boolean **isActive**()

Tests whether the timer MBean is active. A timer MBean is marked active when the [start](http://docs.google.com/javax/management/timer/TimerMBean.html#start()) method is called. It becomes inactive when the [stop](http://docs.google.com/javax/management/timer/TimerMBean.html#stop()) method is called.

**Returns:**true if the timer MBean is active, false otherwise.

### isEmpty

boolean **isEmpty**()

Tests whether the list of timer notifications is empty.

**Returns:**true if the list of timer notifications is empty, false otherwise.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/TimerMBean.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/javax/management/timer/Timer.html)   [**NEXT CLASS**](http://docs.google.com/javax/management/timer/TimerNotification.html) | [**FRAMES**](http://docs.google.com/index.html?javax/management/timer/TimerMBean.html)    [**NO FRAMES**](http://docs.google.com/TimerMBean.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[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).