

[Overview](#) [Package](#) [Class](#) [Use](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)[Prev Class](#) [Next Class](#) [Frames](#) [No Frames](#) [All Classes](#)Summary: [Nested](#) | [Field](#) | [Constr](#) | [Method](#) Detail: [Field](#) | [Constr](#) | [Method](#)

timer

Class Timer

java.lang.Object

timer.Timer

```
public class Timer
extends java.lang.Object
```

Field Summary

Fields

Modifier and Type	Field and Description
static int	PERIODE_DEFAULT

Constructor Summary

Constructors

Constructor and Description
Timer() Constructeur par défaut Initialise un Timer avec : Timer.PERIODE_DEFAULT pour la période Le timer sera inactif
Timer(double periode) Constructeur d'initialisation champ à champ
Timer(Timer unTimer) Constructeur par copie

Method Summary

Methods

Modifier and Type	Method and Description
void	activer() Rend le Timer actif pour une utilisation
void	desactiver() Rend le Timer inactif

boolean	equals (java.lang.Object o) Méthode equals()
boolean	getEtat () Accesseur de l'activité du Timer
double	getPeriode () Accesseur pour la période du Timer
void	init () Méthode d'initialisation interactive
void	setEtat (boolean etat) Mutateur de l'activité du Timer
void	setPeriode (double periode) Mutateur pour la période du Timer
java.lang.String	toString () Méthode toString()
void	utiliser () Utilise le Timer Appel bloquant pour une durée définit par la période du Timer

Methods inherited from class java.lang.Object

getClass, hashCode, notify, notifyAll, wait, wait, wait

Field Detail

PERIODE_DEFAULT

```
public static final int PERIODE_DEFAULT
```

See Also:

[Constant Field Values](#)

Constructor Detail

Timer

```
public Timer()
```

Constructeur par défaut
Initialise un Timer avec :
Timer.PERIODE_DEFAULT pour la période
Le timer sera inactif

Timer

```
public Timer(double periode)
```

Constructeur d'initialisation champ à champ

Parameters:

`periode` - : période en milliseconde

Throws:

`java.lang.Exception` - : si la période est négative

Timer

```
public Timer(Timer unTimer)
```

Constructeur par recopie

Parameters:

`unTimer` -

Method Detail

getEtat

```
public boolean getEtat()
```

Accesseur de l'activité du Timer

Returns:

(boolean) : true, si Timer activé sinon false

setPeriode

```
public void setPeriode(double periode)
```

Mutateur pour la période du Timer

Parameters:

`periode` - (double) : période du Timer en milli-secondes

getPeriode

```
public double getPeriode()
```

Accesseur pour la période du Timer

Returns:

(double) : période du Timer en milli-secondes

setEtat

```
public void setEtat(boolean etat)
```

Mutateur de l'activité du Timer

Parameters:

etat - (boolean) : true, si Timer activé sinon false

activer

```
public void activer()
```

Rend le Timer actif pour une utilisation

desactiver

```
public void desactiver()
```

Rend le Timer inactif

utiliser

```
public void utiliser()
```

Utilise le Timer

Appel bloquant pour une durée définit par la période du Timer

init

```
public void init()
```

Méthode d'initialisation interactive

toString

```
public java.lang.String toString()
```

Méthode toString()

Overrides:

toString in class java.lang.Object

equals

```
public boolean equals(java.lang.Object o)
```

Méthode equals()

Overrides:

equals in class java.lang.Object



[Overview](#) [Package](#) [Class](#) [Use](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[Prev Class](#) [Next Class](#) [Frames](#) [No Frames](#) [All Classes](#)

Summary: [Nested](#) | [Field](#) | [Constr](#) | [Method](#) [Detail: Field](#) | [Constr](#) | [Method](#)
