| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Level.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/util/logging/Handler.html)   [**NEXT CLASS**](http://docs.google.com/java/util/logging/Logger.html) | [**FRAMES**](http://docs.google.com/index.html?java/util/logging/Level.html)    [**NO FRAMES**](http://docs.google.com/Level.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#2jxsxqh) | [METHOD](#1y810tw) |

## **java.util.logging**

Class Level

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **java.util.logging.Level**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html)

public class **Level**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Serializable](http://docs.google.com/java/io/Serializable.html)

The Level class defines a set of standard logging levels that can be used to control logging output. The logging Level objects are ordered and are specified by ordered integers. Enabling logging at a given level also enables logging at all higher levels.

Clients should normally use the predefined Level constants such as Level.SEVERE.

The levels in descending order are:

* SEVERE (highest value)
* WARNING
* INFO
* CONFIG
* FINE
* FINER
* FINEST (lowest value)

In addition there is a level OFF that can be used to turn off logging, and a level ALL that can be used to enable logging of all messages.

It is possible for third parties to define additional logging levels by subclassing Level. In such cases subclasses should take care to chose unique integer level values and to ensure that they maintain the Object uniqueness property across serialization by defining a suitable readResolve method.

**Since:** 1.4 **See Also:**[Serialized Form](http://docs.google.com/serialized-form.html#java.util.logging.Level)

| **Field Summary** | |
| --- | --- |
| static [Level](http://docs.google.com/java/util/logging/Level.html) | [**ALL**](http://docs.google.com/java/util/logging/Level.html#ALL)            ALL indicates that all messages should be logged. |
| static [Level](http://docs.google.com/java/util/logging/Level.html) | [**CONFIG**](http://docs.google.com/java/util/logging/Level.html#CONFIG)            CONFIG is a message level for static configuration messages. |
| static [Level](http://docs.google.com/java/util/logging/Level.html) | [**FINE**](http://docs.google.com/java/util/logging/Level.html#FINE)            FINE is a message level providing tracing information. |
| static [Level](http://docs.google.com/java/util/logging/Level.html) | [**FINER**](http://docs.google.com/java/util/logging/Level.html#FINER)            FINER indicates a fairly detailed tracing message. |
| static [Level](http://docs.google.com/java/util/logging/Level.html) | [**FINEST**](http://docs.google.com/java/util/logging/Level.html#FINEST)            FINEST indicates a highly detailed tracing message. |
| static [Level](http://docs.google.com/java/util/logging/Level.html) | [**INFO**](http://docs.google.com/java/util/logging/Level.html#INFO)            INFO is a message level for informational messages. |
| static [Level](http://docs.google.com/java/util/logging/Level.html) | [**OFF**](http://docs.google.com/java/util/logging/Level.html#OFF)            OFF is a special level that can be used to turn off logging. |
| static [Level](http://docs.google.com/java/util/logging/Level.html) | [**SEVERE**](http://docs.google.com/java/util/logging/Level.html#SEVERE)            SEVERE is a message level indicating a serious failure. |
| static [Level](http://docs.google.com/java/util/logging/Level.html) | [**WARNING**](http://docs.google.com/java/util/logging/Level.html#WARNING)            WARNING is a message level indicating a potential problem. |

| **Constructor Summary** | |
| --- | --- |
| protected | [**Level**](http://docs.google.com/java/util/logging/Level.html#Level(java.lang.String,%20int))([String](http://docs.google.com/java/lang/String.html) name, int value)            Create a named Level with a given integer value. |
| protected | [**Level**](http://docs.google.com/java/util/logging/Level.html#Level(java.lang.String,%20int,%20java.lang.String))([String](http://docs.google.com/java/lang/String.html) name, int value, [String](http://docs.google.com/java/lang/String.html) resourceBundleName)            Create a named Level with a given integer value and a given localization resource name. |

| **Method Summary** | |
| --- | --- |
| boolean | [**equals**](http://docs.google.com/java/util/logging/Level.html#equals(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) ox)            Compare two objects for value equality. |
| [String](http://docs.google.com/java/lang/String.html) | [**getLocalizedName**](http://docs.google.com/java/util/logging/Level.html#getLocalizedName())()            Return the localized string name of the Level, for the current default locale. |
| [String](http://docs.google.com/java/lang/String.html) | [**getName**](http://docs.google.com/java/util/logging/Level.html#getName())()            Return the non-localized string name of the Level. |
| [String](http://docs.google.com/java/lang/String.html) | [**getResourceBundleName**](http://docs.google.com/java/util/logging/Level.html#getResourceBundleName())()            Return the level's localization resource bundle name, or null if no localization bundle is defined. |
| int | [**hashCode**](http://docs.google.com/java/util/logging/Level.html#hashCode())()            Generate a hashcode. |
| int | [**intValue**](http://docs.google.com/java/util/logging/Level.html#intValue())()            Get the integer value for this level. |
| static [Level](http://docs.google.com/java/util/logging/Level.html) | [**parse**](http://docs.google.com/java/util/logging/Level.html#parse(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Parse a level name string into a Level. |
| [String](http://docs.google.com/java/lang/String.html) | [**toString**](http://docs.google.com/java/util/logging/Level.html#toString())()            Returns a string representation of the object. |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [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)) |

| **Field Detail** |
| --- |

### OFF

public static final [Level](http://docs.google.com/java/util/logging/Level.html) **OFF**

OFF is a special level that can be used to turn off logging. This level is initialized to Integer.MAX\_VALUE.

### SEVERE

public static final [Level](http://docs.google.com/java/util/logging/Level.html) **SEVERE**

SEVERE is a message level indicating a serious failure.

In general SEVERE messages should describe events that are of considerable importance and which will prevent normal program execution. They should be reasonably intelligible to end users and to system administrators. This level is initialized to 1000.

### WARNING

public static final [Level](http://docs.google.com/java/util/logging/Level.html) **WARNING**

WARNING is a message level indicating a potential problem.

In general WARNING messages should describe events that will be of interest to end users or system managers, or which indicate potential problems. This level is initialized to 900.

### INFO

public static final [Level](http://docs.google.com/java/util/logging/Level.html) **INFO**

INFO is a message level for informational messages.

Typically INFO messages will be written to the console or its equivalent. So the INFO level should only be used for reasonably significant messages that will make sense to end users and system admins. This level is initialized to 800.

### CONFIG

public static final [Level](http://docs.google.com/java/util/logging/Level.html) **CONFIG**

CONFIG is a message level for static configuration messages.

CONFIG messages are intended to provide a variety of static configuration information, to assist in debugging problems that may be associated with particular configurations. For example, CONFIG message might include the CPU type, the graphics depth, the GUI look-and-feel, etc. This level is initialized to 700.

### FINE

public static final [Level](http://docs.google.com/java/util/logging/Level.html) **FINE**

FINE is a message level providing tracing information.

All of FINE, FINER, and FINEST are intended for relatively detailed tracing. The exact meaning of the three levels will vary between subsystems, but in general, FINEST should be used for the most voluminous detailed output, FINER for somewhat less detailed output, and FINE for the lowest volume (and most important) messages.

In general the FINE level should be used for information that will be broadly interesting to developers who do not have a specialized interest in the specific subsystem.

FINE messages might include things like minor (recoverable) failures. Issues indicating potential performance problems are also worth logging as FINE. This level is initialized to 500.

### FINER

public static final [Level](http://docs.google.com/java/util/logging/Level.html) **FINER**

FINER indicates a fairly detailed tracing message. By default logging calls for entering, returning, or throwing an exception are traced at this level. This level is initialized to 400.

### FINEST

public static final [Level](http://docs.google.com/java/util/logging/Level.html) **FINEST**

FINEST indicates a highly detailed tracing message. This level is initialized to 300.

### ALL

public static final [Level](http://docs.google.com/java/util/logging/Level.html) **ALL**

ALL indicates that all messages should be logged. This level is initialized to Integer.MIN\_VALUE.

| **Constructor Detail** |
| --- |

### Level

protected **Level**([String](http://docs.google.com/java/lang/String.html) name,  
 int value)

Create a named Level with a given integer value.

Note that this constructor is "protected" to allow subclassing. In general clients of logging should use one of the constant Level objects such as SEVERE or FINEST. However, if clients need to add new logging levels, they may subclass Level and define new constants.

**Parameters:**name - the name of the Level, for example "SEVERE".value - an integer value for the level. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the name is null

### Level

protected **Level**([String](http://docs.google.com/java/lang/String.html) name,  
 int value,  
 [String](http://docs.google.com/java/lang/String.html) resourceBundleName)

Create a named Level with a given integer value and a given localization resource name.

**Parameters:**name - the name of the Level, for example "SEVERE".value - an integer value for the level.resourceBundleName - name of a resource bundle to use in localizing the given name. If the resourceBundleName is null or an empty string, it is ignored. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the name is null

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

### getResourceBundleName

public [String](http://docs.google.com/java/lang/String.html) **getResourceBundleName**()

Return the level's localization resource bundle name, or null if no localization bundle is defined.

**Returns:**localization resource bundle name

### getName

public [String](http://docs.google.com/java/lang/String.html) **getName**()

Return the non-localized string name of the Level.

**Returns:**non-localized name

### getLocalizedName

public [String](http://docs.google.com/java/lang/String.html) **getLocalizedName**()

Return the localized string name of the Level, for the current default locale.

If no localization information is available, the non-localized name is returned.

**Returns:**localized name

### toString

public final [String](http://docs.google.com/java/lang/String.html) **toString**()

**Description copied from class:** [**Object**](http://docs.google.com/java/lang/Object.html#toString()) Returns a string representation of the object. In general, the toString method returns a string that "textually represents" this object. The result should be a concise but informative representation that is easy for a person to read. It is recommended that all subclasses override this method.

The toString method for class Object returns a string consisting of the name of the class of which the object is an instance, the at-sign character `@', and the unsigned hexadecimal representation of the hash code of the object. In other words, this method returns a string equal to the value of:

getClass().getName() + '@' + Integer.toHexString(hashCode())

**Overrides:**[toString](http://docs.google.com/java/lang/Object.html#toString()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**the non-localized name of the Level, for example "INFO".

### intValue

public final int **intValue**()

Get the integer value for this level. This integer value can be used for efficient ordering comparisons between Level objects.

**Returns:**the integer value for this level.

### parse

public static [Level](http://docs.google.com/java/util/logging/Level.html) **parse**([String](http://docs.google.com/java/lang/String.html) name)  
 throws [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html)

Parse a level name string into a Level.

The argument string may consist of either a level name or an integer value.

For example:

* "SEVERE"
* "1000"

**Parameters:**name - string to be parsed **Returns:**The parsed value. Passing an integer that corresponds to a known name (eg 700) will return the associated name (eg CONFIG). Passing an integer that does not (eg 1) will return a new level name initialized to that value. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the name is null [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the value is not valid. Valid values are integers between Integer.MIN\_VALUE and Integer.MAX\_VALUE, and all known level names. Known names are the levels defined by this class (i.e. FINE, FINER, FINEST), or created by this class with appropriate package access, or new levels defined or created by subclasses.

### equals

public boolean **equals**([Object](http://docs.google.com/java/lang/Object.html) ox)

Compare two objects for value equality.

**Overrides:**[equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)) in class [Object](http://docs.google.com/java/lang/Object.html) **Parameters:**ox - the reference object with which to compare. **Returns:**true if and only if the two objects have the same level value.**See Also:**[Object.hashCode()](http://docs.google.com/java/lang/Object.html#hashCode()), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### hashCode

public int **hashCode**()

Generate a hashcode.

**Overrides:**[hashCode](http://docs.google.com/java/lang/Object.html#hashCode()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a hashcode based on the level value**See Also:**[Object.equals(java.lang.Object)](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Level.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/util/logging/Handler.html)   [**NEXT CLASS**](http://docs.google.com/java/util/logging/Logger.html) | [**FRAMES**](http://docs.google.com/index.html?java/util/logging/Level.html)    [**NO FRAMES**](http://docs.google.com/Level.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#2jxsxqh) | [METHOD](#1y810tw) |

[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).