| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ZipFile.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/zip/ZipException.html)   [**NEXT CLASS**](http://docs.google.com/java/util/zip/ZipInputStream.html) | [**FRAMES**](http://docs.google.com/index.html?java/util/zip/ZipFile.html)    [**NO FRAMES**](http://docs.google.com/ZipFile.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#46r0co2) | [METHOD](#206ipza) |

## **java.util.zip**

Class ZipFile

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

**Direct Known Subclasses:** [JarFile](http://docs.google.com/java/util/jar/JarFile.html)

public class **ZipFile**extends [Object](http://docs.google.com/java/lang/Object.html)

This class is used to read entries from a zip file.

Unless otherwise noted, passing a null argument to a constructor or method in this class will cause a [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) to be thrown.

| **Field Summary** | |
| --- | --- |
| static int | [**CENATT**](http://docs.google.com/java/util/zip/ZipFile.html#CENATT) |
| static int | [**CENATX**](http://docs.google.com/java/util/zip/ZipFile.html#CENATX) |
| static int | [**CENCOM**](http://docs.google.com/java/util/zip/ZipFile.html#CENCOM) |
| static int | [**CENCRC**](http://docs.google.com/java/util/zip/ZipFile.html#CENCRC) |
| static int | [**CENDSK**](http://docs.google.com/java/util/zip/ZipFile.html#CENDSK) |
| static int | [**CENEXT**](http://docs.google.com/java/util/zip/ZipFile.html#CENEXT) |
| static int | [**CENFLG**](http://docs.google.com/java/util/zip/ZipFile.html#CENFLG) |
| static int | [**CENHDR**](http://docs.google.com/java/util/zip/ZipFile.html#CENHDR) |
| static int | [**CENHOW**](http://docs.google.com/java/util/zip/ZipFile.html#CENHOW) |
| static int | [**CENLEN**](http://docs.google.com/java/util/zip/ZipFile.html#CENLEN) |
| static int | [**CENNAM**](http://docs.google.com/java/util/zip/ZipFile.html#CENNAM) |
| static int | [**CENOFF**](http://docs.google.com/java/util/zip/ZipFile.html#CENOFF) |
| static long | [**CENSIG**](http://docs.google.com/java/util/zip/ZipFile.html#CENSIG) |
| static int | [**CENSIZ**](http://docs.google.com/java/util/zip/ZipFile.html#CENSIZ) |
| static int | [**CENTIM**](http://docs.google.com/java/util/zip/ZipFile.html#CENTIM) |
| static int | [**CENVEM**](http://docs.google.com/java/util/zip/ZipFile.html#CENVEM) |
| static int | [**CENVER**](http://docs.google.com/java/util/zip/ZipFile.html#CENVER) |
| static int | [**ENDCOM**](http://docs.google.com/java/util/zip/ZipFile.html#ENDCOM) |
| static int | [**ENDHDR**](http://docs.google.com/java/util/zip/ZipFile.html#ENDHDR) |
| static int | [**ENDOFF**](http://docs.google.com/java/util/zip/ZipFile.html#ENDOFF) |
| static long | [**ENDSIG**](http://docs.google.com/java/util/zip/ZipFile.html#ENDSIG) |
| static int | [**ENDSIZ**](http://docs.google.com/java/util/zip/ZipFile.html#ENDSIZ) |
| static int | [**ENDSUB**](http://docs.google.com/java/util/zip/ZipFile.html#ENDSUB) |
| static int | [**ENDTOT**](http://docs.google.com/java/util/zip/ZipFile.html#ENDTOT) |
| static int | [**EXTCRC**](http://docs.google.com/java/util/zip/ZipFile.html#EXTCRC) |
| static int | [**EXTHDR**](http://docs.google.com/java/util/zip/ZipFile.html#EXTHDR) |
| static int | [**EXTLEN**](http://docs.google.com/java/util/zip/ZipFile.html#EXTLEN) |
| static long | [**EXTSIG**](http://docs.google.com/java/util/zip/ZipFile.html#EXTSIG) |
| static int | [**EXTSIZ**](http://docs.google.com/java/util/zip/ZipFile.html#EXTSIZ) |
| static int | [**LOCCRC**](http://docs.google.com/java/util/zip/ZipFile.html#LOCCRC) |
| static int | [**LOCEXT**](http://docs.google.com/java/util/zip/ZipFile.html#LOCEXT) |
| static int | [**LOCFLG**](http://docs.google.com/java/util/zip/ZipFile.html#LOCFLG) |
| static int | [**LOCHDR**](http://docs.google.com/java/util/zip/ZipFile.html#LOCHDR) |
| static int | [**LOCHOW**](http://docs.google.com/java/util/zip/ZipFile.html#LOCHOW) |
| static int | [**LOCLEN**](http://docs.google.com/java/util/zip/ZipFile.html#LOCLEN) |
| static int | [**LOCNAM**](http://docs.google.com/java/util/zip/ZipFile.html#LOCNAM) |
| static long | [**LOCSIG**](http://docs.google.com/java/util/zip/ZipFile.html#LOCSIG) |
| static int | [**LOCSIZ**](http://docs.google.com/java/util/zip/ZipFile.html#LOCSIZ) |
| static int | [**LOCTIM**](http://docs.google.com/java/util/zip/ZipFile.html#LOCTIM) |
| static int | [**LOCVER**](http://docs.google.com/java/util/zip/ZipFile.html#LOCVER) |
| static int | [**OPEN\_DELETE**](http://docs.google.com/java/util/zip/ZipFile.html#OPEN_DELETE)            Mode flag to open a zip file and mark it for deletion. |
| static int | [**OPEN\_READ**](http://docs.google.com/java/util/zip/ZipFile.html#OPEN_READ)            Mode flag to open a zip file for reading. |

| **Constructor Summary** | |
| --- | --- |
| [**ZipFile**](http://docs.google.com/java/util/zip/ZipFile.html#ZipFile(java.io.File))([File](http://docs.google.com/java/io/File.html) file)            Opens a ZIP file for reading given the specified File object. |
| [**ZipFile**](http://docs.google.com/java/util/zip/ZipFile.html#ZipFile(java.io.File,%20int))([File](http://docs.google.com/java/io/File.html) file, int mode)            Opens a new ZipFile to read from the specified File object in the specified mode. |
| [**ZipFile**](http://docs.google.com/java/util/zip/ZipFile.html#ZipFile(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Opens a zip file for reading. |

| **Method Summary** | |
| --- | --- |
| void | [**close**](http://docs.google.com/java/util/zip/ZipFile.html#close())()            Closes the ZIP file. |
| [Enumeration](http://docs.google.com/java/util/Enumeration.html)<? extends [ZipEntry](http://docs.google.com/java/util/zip/ZipEntry.html)> | [**entries**](http://docs.google.com/java/util/zip/ZipFile.html#entries())()            Returns an enumeration of the ZIP file entries. |
| protected  void | [**finalize**](http://docs.google.com/java/util/zip/ZipFile.html#finalize())()            Ensures that the close method of this ZIP file is called when there are no more references to it. |
| [ZipEntry](http://docs.google.com/java/util/zip/ZipEntry.html) | [**getEntry**](http://docs.google.com/java/util/zip/ZipFile.html#getEntry(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Returns the zip file entry for the specified name, or null if not found. |
| [InputStream](http://docs.google.com/java/io/InputStream.html) | [**getInputStream**](http://docs.google.com/java/util/zip/ZipFile.html#getInputStream(java.util.zip.ZipEntry))([ZipEntry](http://docs.google.com/java/util/zip/ZipEntry.html) entry)            Returns an input stream for reading the contents of the specified zip file entry. |
| [String](http://docs.google.com/java/lang/String.html) | [**getName**](http://docs.google.com/java/util/zip/ZipFile.html#getName())()            Returns the path name of the ZIP file. |
| int | [**size**](http://docs.google.com/java/util/zip/ZipFile.html#size())()            Returns the number of entries in the ZIP file. |

| **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()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [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** |
| --- |

### OPEN\_READ

public static final int **OPEN\_READ**

Mode flag to open a zip file for reading.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.OPEN_READ)

### OPEN\_DELETE

public static final int **OPEN\_DELETE**

Mode flag to open a zip file and mark it for deletion. The file will be deleted some time between the moment that it is opened and the moment that it is closed, but its contents will remain accessible via the ZipFile object until either the close method is invoked or the virtual machine exits.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.OPEN_DELETE)

### LOCSIG

public static final long **LOCSIG**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.LOCSIG)

### EXTSIG

public static final long **EXTSIG**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.EXTSIG)

### CENSIG

public static final long **CENSIG**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENSIG)

### ENDSIG

public static final long **ENDSIG**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.ENDSIG)

### LOCHDR

public static final int **LOCHDR**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.LOCHDR)

### EXTHDR

public static final int **EXTHDR**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.EXTHDR)

### CENHDR

public static final int **CENHDR**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENHDR)

### ENDHDR

public static final int **ENDHDR**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.ENDHDR)

### LOCVER

public static final int **LOCVER**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.LOCVER)

### LOCFLG

public static final int **LOCFLG**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.LOCFLG)

### LOCHOW

public static final int **LOCHOW**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.LOCHOW)

### LOCTIM

public static final int **LOCTIM**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.LOCTIM)

### LOCCRC

public static final int **LOCCRC**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.LOCCRC)

### LOCSIZ

public static final int **LOCSIZ**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.LOCSIZ)

### LOCLEN

public static final int **LOCLEN**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.LOCLEN)

### LOCNAM

public static final int **LOCNAM**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.LOCNAM)

### LOCEXT

public static final int **LOCEXT**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.LOCEXT)

### EXTCRC

public static final int **EXTCRC**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.EXTCRC)

### EXTSIZ

public static final int **EXTSIZ**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.EXTSIZ)

### EXTLEN

public static final int **EXTLEN**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.EXTLEN)

### CENVEM

public static final int **CENVEM**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENVEM)

### CENVER

public static final int **CENVER**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENVER)

### CENFLG

public static final int **CENFLG**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENFLG)

### CENHOW

public static final int **CENHOW**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENHOW)

### CENTIM

public static final int **CENTIM**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENTIM)

### CENCRC

public static final int **CENCRC**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENCRC)

### CENSIZ

public static final int **CENSIZ**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENSIZ)

### CENLEN

public static final int **CENLEN**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENLEN)

### CENNAM

public static final int **CENNAM**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENNAM)

### CENEXT

public static final int **CENEXT**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENEXT)

### CENCOM

public static final int **CENCOM**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENCOM)

### CENDSK

public static final int **CENDSK**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENDSK)

### CENATT

public static final int **CENATT**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENATT)

### CENATX

public static final int **CENATX**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENATX)

### CENOFF

public static final int **CENOFF**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.CENOFF)

### ENDSUB

public static final int **ENDSUB**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.ENDSUB)

### ENDTOT

public static final int **ENDTOT**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.ENDTOT)

### ENDSIZ

public static final int **ENDSIZ**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.ENDSIZ)

### ENDOFF

public static final int **ENDOFF**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.ENDOFF)

### ENDCOM

public static final int **ENDCOM**

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.util.zip.ZipFile.ENDCOM)

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

### ZipFile

public **ZipFile**([String](http://docs.google.com/java/lang/String.html) name)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Opens a zip file for reading.

First, if there is a security manager, its checkRead method is called with the name argument as its argument to ensure the read is allowed.

**Parameters:**name - the name of the zip file **Throws:** [ZipException](http://docs.google.com/java/util/zip/ZipException.html) - if a ZIP format error has occurred [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error has occurred [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a security manager exists and its checkRead method doesn't allow read access to the file.**See Also:**[SecurityManager.checkRead(java.lang.String)](http://docs.google.com/java/lang/SecurityManager.html#checkRead(java.lang.String))

### ZipFile

public **ZipFile**([File](http://docs.google.com/java/io/File.html) file,  
 int mode)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Opens a new ZipFile to read from the specified File object in the specified mode. The mode argument must be either OPEN\_READ or OPEN\_READ | OPEN\_DELETE.

First, if there is a security manager, its checkRead method is called with the name argument as its argument to ensure the read is allowed.

**Parameters:**file - the ZIP file to be opened for readingmode - the mode in which the file is to be opened **Throws:** [ZipException](http://docs.google.com/java/util/zip/ZipException.html) - if a ZIP format error has occurred [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error has occurred [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a security manager exists and its checkRead method doesn't allow read access to the file, or its checkDelete method doesn't allow deleting the file when the OPEN\_DELETE flag is set. [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the mode argument is invalid**Since:** 1.3 **See Also:**[SecurityManager.checkRead(java.lang.String)](http://docs.google.com/java/lang/SecurityManager.html#checkRead(java.lang.String))

### ZipFile

public **ZipFile**([File](http://docs.google.com/java/io/File.html) file)  
 throws [ZipException](http://docs.google.com/java/util/zip/ZipException.html),  
 [IOException](http://docs.google.com/java/io/IOException.html)

Opens a ZIP file for reading given the specified File object.

**Parameters:**file - the ZIP file to be opened for reading **Throws:** [ZipException](http://docs.google.com/java/util/zip/ZipException.html) - if a ZIP error has occurred [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error has occurred

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

### getEntry

public [ZipEntry](http://docs.google.com/java/util/zip/ZipEntry.html) **getEntry**([String](http://docs.google.com/java/lang/String.html) name)

Returns the zip file entry for the specified name, or null if not found.

**Parameters:**name - the name of the entry **Returns:**the zip file entry, or null if not found **Throws:** [IllegalStateException](http://docs.google.com/java/lang/IllegalStateException.html) - if the zip file has been closed

### getInputStream

public [InputStream](http://docs.google.com/java/io/InputStream.html) **getInputStream**([ZipEntry](http://docs.google.com/java/util/zip/ZipEntry.html) entry)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Returns an input stream for reading the contents of the specified zip file entry.

Closing this ZIP file will, in turn, close all input streams that have been returned by invocations of this method.

**Parameters:**entry - the zip file entry **Returns:**the input stream for reading the contents of the specified zip file entry. **Throws:** [ZipException](http://docs.google.com/java/util/zip/ZipException.html) - if a ZIP format error has occurred [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error has occurred [IllegalStateException](http://docs.google.com/java/lang/IllegalStateException.html) - if the zip file has been closed

### getName

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

Returns the path name of the ZIP file.

**Returns:**the path name of the ZIP file

### entries

public [Enumeration](http://docs.google.com/java/util/Enumeration.html)<? extends [ZipEntry](http://docs.google.com/java/util/zip/ZipEntry.html)> **entries**()

Returns an enumeration of the ZIP file entries.

**Returns:**an enumeration of the ZIP file entries **Throws:** [IllegalStateException](http://docs.google.com/java/lang/IllegalStateException.html) - if the zip file has been closed

### size

public int **size**()

Returns the number of entries in the ZIP file.

**Returns:**the number of entries in the ZIP file **Throws:** [IllegalStateException](http://docs.google.com/java/lang/IllegalStateException.html) - if the zip file has been closed

### close

public void **close**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Closes the ZIP file.

Closing this ZIP file will close all of the input streams previously returned by invocations of the [getInputStream](http://docs.google.com/java/util/zip/ZipFile.html#getInputStream(java.util.zip.ZipEntry)) method.

**Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error has occurred

### finalize

protected void **finalize**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Ensures that the close method of this ZIP file is called when there are no more references to it.

Since the time when GC would invoke this method is undetermined, it is strongly recommended that applications invoke the close method as soon they have finished accessing this ZipFile. This will prevent holding up system resources for an undetermined length of time.

**Overrides:**[finalize](http://docs.google.com/java/lang/Object.html#finalize()) in class [Object](http://docs.google.com/java/lang/Object.html) **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error has occurred**See Also:**[close()](http://docs.google.com/java/util/zip/ZipFile.html#close())

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ZipFile.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/zip/ZipException.html)   [**NEXT CLASS**](http://docs.google.com/java/util/zip/ZipInputStream.html) | [**FRAMES**](http://docs.google.com/index.html?java/util/zip/ZipFile.html)    [**NO FRAMES**](http://docs.google.com/ZipFile.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#46r0co2) | [METHOD](#206ipza) |

[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).