| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/LineNumberInputStream.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/io/IOException.html)   [**NEXT CLASS**](http://docs.google.com/java/io/LineNumberReader.html) | [**FRAMES**](http://docs.google.com/index.html?java/io/LineNumberInputStream.html)    [**NO FRAMES**](http://docs.google.com/LineNumberInputStream.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#3rdcrjn) |

## **java.io**

Class LineNumberInputStream

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.io.InputStream](http://docs.google.com/java/io/InputStream.html)  
 [java.io.FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html)  
 **java.io.LineNumberInputStream**

**All Implemented Interfaces:** [Closeable](http://docs.google.com/java/io/Closeable.html)**Deprecated.** *This class incorrectly assumes that bytes adequately represent characters. As of JDK 1.1, the preferred way to operate on character streams is via the new character-stream classes, which include a class for counting line numbers.*

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
public class **LineNumberInputStream**extends [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html)

This class is an input stream filter that provides the added functionality of keeping track of the current line number.

A line is a sequence of bytes ending with a carriage return character ('\r'), a newline character ('\n'), or a carriage return character followed immediately by a linefeed character. In all three cases, the line terminating character(s) are returned as a single newline character.

The line number begins at 0, and is incremented by 1 when a read returns a newline character.

**Since:** JDK1.0 **See Also:**[LineNumberReader](http://docs.google.com/java/io/LineNumberReader.html)

| **Field Summary** | |
| --- | --- |

| **Fields inherited from class java.io.**[**FilterInputStream**](http://docs.google.com/java/io/FilterInputStream.html) |
| --- |
| [in](http://docs.google.com/java/io/FilterInputStream.html#in) |

| **Constructor Summary** | |
| --- | --- |
| [**LineNumberInputStream**](http://docs.google.com/java/io/LineNumberInputStream.html#LineNumberInputStream(java.io.InputStream))([InputStream](http://docs.google.com/java/io/InputStream.html) in)  **Deprecated.** Constructs a newline number input stream that reads its input from the specified input stream. |

| **Method Summary** | |
| --- | --- |
| int | [**available**](http://docs.google.com/java/io/LineNumberInputStream.html#available())()  **Deprecated.** Returns the number of bytes that can be read from this input stream without blocking. |
| int | [**getLineNumber**](http://docs.google.com/java/io/LineNumberInputStream.html#getLineNumber())()  **Deprecated.** Returns the current line number. |
| void | [**mark**](http://docs.google.com/java/io/LineNumberInputStream.html#mark(int))(int readlimit)  **Deprecated.** Marks the current position in this input stream. |
| int | [**read**](http://docs.google.com/java/io/LineNumberInputStream.html#read())()  **Deprecated.** Reads the next byte of data from this input stream. |
| int | [**read**](http://docs.google.com/java/io/LineNumberInputStream.html#read(byte%5B%5D,%20int,%20int))(byte[] b, int off, int len)  **Deprecated.** Reads up to len bytes of data from this input stream into an array of bytes. |
| void | [**reset**](http://docs.google.com/java/io/LineNumberInputStream.html#reset())()  **Deprecated.** Repositions this stream to the position at the time the mark method was last called on this input stream. |
| void | [**setLineNumber**](http://docs.google.com/java/io/LineNumberInputStream.html#setLineNumber(int))(int lineNumber)  **Deprecated.** Sets the line number to the specified argument. |
| long | [**skip**](http://docs.google.com/java/io/LineNumberInputStream.html#skip(long))(long n)  **Deprecated.** Skips over and discards n bytes of data from this input stream. |

| **Methods inherited from class java.io.**[**FilterInputStream**](http://docs.google.com/java/io/FilterInputStream.html) |
| --- |
| [close](http://docs.google.com/java/io/FilterInputStream.html#close()), [markSupported](http://docs.google.com/java/io/FilterInputStream.html#markSupported()), [read](http://docs.google.com/java/io/FilterInputStream.html#read(byte%5B%5D)) |

| **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)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [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)) |

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

### LineNumberInputStream

public **LineNumberInputStream**([InputStream](http://docs.google.com/java/io/InputStream.html) in)

**Deprecated.** Constructs a newline number input stream that reads its input from the specified input stream.

**Parameters:**in - the underlying input stream.

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

### read

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

**Deprecated.** Reads the next byte of data from this input stream. The value byte is returned as an int in the range 0 to 255. If no byte is available because the end of the stream has been reached, the value -1 is returned. This method blocks until input data is available, the end of the stream is detected, or an exception is thrown.

The read method of LineNumberInputStream calls the read method of the underlying input stream. It checks for carriage returns and newline characters in the input, and modifies the current line number as appropriate. A carriage-return character or a carriage return followed by a newline character are both converted into a single newline character.

**Overrides:**[read](http://docs.google.com/java/io/FilterInputStream.html#read()) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Returns:**the next byte of data, or -1 if the end of this stream is reached. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.**See Also:**[FilterInputStream.in](http://docs.google.com/java/io/FilterInputStream.html#in), [getLineNumber()](http://docs.google.com/java/io/LineNumberInputStream.html#getLineNumber())

### read

public int **read**(byte[] b,  
 int off,  
 int len)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

**Deprecated.** Reads up to len bytes of data from this input stream into an array of bytes. This method blocks until some input is available.

The read method of LineNumberInputStream repeatedly calls the read method of zero arguments to fill in the byte array.

**Overrides:**[read](http://docs.google.com/java/io/FilterInputStream.html#read(byte%5B%5D,%20int,%20int)) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Parameters:**b - the buffer into which the data is read.off - the start offset of the data.len - the maximum number of bytes read. **Returns:**the total number of bytes read into the buffer, or -1 if there is no more data because the end of this stream has been reached. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.**See Also:**[read()](http://docs.google.com/java/io/LineNumberInputStream.html#read())

### skip

public long **skip**(long n)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

**Deprecated.** Skips over and discards n bytes of data from this input stream. The skip method may, for a variety of reasons, end up skipping over some smaller number of bytes, possibly 0. The actual number of bytes skipped is returned. If n is negative, no bytes are skipped.

The skip method of LineNumberInputStream creates a byte array and then repeatedly reads into it until n bytes have been read or the end of the stream has been reached.

**Overrides:**[skip](http://docs.google.com/java/io/FilterInputStream.html#skip(long)) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Parameters:**n - the number of bytes to be skipped. **Returns:**the actual number of bytes skipped. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.**See Also:**[FilterInputStream.in](http://docs.google.com/java/io/FilterInputStream.html#in)

### setLineNumber

public void **setLineNumber**(int lineNumber)

**Deprecated.** Sets the line number to the specified argument.

**Parameters:**lineNumber - the new line number.**See Also:**[getLineNumber()](http://docs.google.com/java/io/LineNumberInputStream.html#getLineNumber())

### getLineNumber

public int **getLineNumber**()

**Deprecated.** Returns the current line number.

**Returns:**the current line number.**See Also:**[setLineNumber(int)](http://docs.google.com/java/io/LineNumberInputStream.html#setLineNumber(int))

### available

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

**Deprecated.** Returns the number of bytes that can be read from this input stream without blocking.

Note that if the underlying input stream is able to supply *k* input characters without blocking, the LineNumberInputStream can guarantee only to provide *k*/2 characters without blocking, because the *k* characters from the underlying input stream might consist of *k*/2 pairs of '\r' and '\n', which are converted to just *k*/2 '\n' characters.

**Overrides:**[available](http://docs.google.com/java/io/FilterInputStream.html#available()) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Returns:**the number of bytes that can be read from this input stream without blocking. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.**See Also:**[FilterInputStream.in](http://docs.google.com/java/io/FilterInputStream.html#in)

### mark

public void **mark**(int readlimit)

**Deprecated.** Marks the current position in this input stream. A subsequent call to the reset method repositions this stream at the last marked position so that subsequent reads re-read the same bytes.

The mark method of LineNumberInputStream remembers the current line number in a private variable, and then calls the mark method of the underlying input stream.

**Overrides:**[mark](http://docs.google.com/java/io/FilterInputStream.html#mark(int)) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Parameters:**readlimit - the maximum limit of bytes that can be read before the mark position becomes invalid.**See Also:**[FilterInputStream.in](http://docs.google.com/java/io/FilterInputStream.html#in), [reset()](http://docs.google.com/java/io/LineNumberInputStream.html#reset())

### reset

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

**Deprecated.** Repositions this stream to the position at the time the mark method was last called on this input stream.

The reset method of LineNumberInputStream resets the line number to be the line number at the time the mark method was called, and then calls the reset method of the underlying input stream.

Stream marks are intended to be used in situations where you need to read ahead a little to see what's in the stream. Often this is most easily done by invoking some general parser. If the stream is of the type handled by the parser, it just chugs along happily. If the stream is not of that type, the parser should toss an exception when it fails, which, if it happens within readlimit bytes, allows the outer code to reset the stream and try another parser.

**Overrides:**[reset](http://docs.google.com/java/io/FilterInputStream.html#reset()) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.**See Also:**[FilterInputStream.in](http://docs.google.com/java/io/FilterInputStream.html#in), [mark(int)](http://docs.google.com/java/io/LineNumberInputStream.html#mark(int))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/LineNumberInputStream.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/io/IOException.html)   [**NEXT CLASS**](http://docs.google.com/java/io/LineNumberReader.html) | [**FRAMES**](http://docs.google.com/index.html?java/io/LineNumberInputStream.html)    [**NO FRAMES**](http://docs.google.com/LineNumberInputStream.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#3rdcrjn) |

[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).