| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BufferedInputStream.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   [**NEXT CLASS**](http://docs.google.com/java/io/BufferedOutputStream.html) | [**FRAMES**](http://docs.google.com/index.html?java/io/BufferedInputStream.html)    [**NO FRAMES**](http://docs.google.com/BufferedInputStream.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: [FIELD](#2s8eyo1) | [CONSTR](#1ksv4uv) | [METHOD](#z337ya) |

## **java.io**

Class BufferedInputStream

[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.BufferedInputStream**

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

public class **BufferedInputStream**extends [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html)

A BufferedInputStream adds functionality to another input stream-namely, the ability to buffer the input and to support the mark and reset methods. When the BufferedInputStream is created, an internal buffer array is created. As bytes from the stream are read or skipped, the internal buffer is refilled as necessary from the contained input stream, many bytes at a time. The mark operation remembers a point in the input stream and the reset operation causes all the bytes read since the most recent mark operation to be reread before new bytes are taken from the contained input stream.

**Since:** JDK1.0

| **Field Summary** | |
| --- | --- |
| protected  byte[] | [**buf**](http://docs.google.com/java/io/BufferedInputStream.html#buf)            The internal buffer array where the data is stored. |
| protected  int | [**count**](http://docs.google.com/java/io/BufferedInputStream.html#count)            The index one greater than the index of the last valid byte in the buffer. |
| protected  int | [**marklimit**](http://docs.google.com/java/io/BufferedInputStream.html#marklimit)            The maximum read ahead allowed after a call to the mark method before subsequent calls to the reset method fail. |
| protected  int | [**markpos**](http://docs.google.com/java/io/BufferedInputStream.html#markpos)            The value of the pos field at the time the last mark method was called. |
| protected  int | [**pos**](http://docs.google.com/java/io/BufferedInputStream.html#pos)            The current position in the buffer. |

| **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** | |
| --- | --- |
| [**BufferedInputStream**](http://docs.google.com/java/io/BufferedInputStream.html#BufferedInputStream(java.io.InputStream))([InputStream](http://docs.google.com/java/io/InputStream.html) in)            Creates a BufferedInputStream and saves its argument, the input stream in, for later use. |
| [**BufferedInputStream**](http://docs.google.com/java/io/BufferedInputStream.html#BufferedInputStream(java.io.InputStream,%20int))([InputStream](http://docs.google.com/java/io/InputStream.html) in, int size)            Creates a BufferedInputStream with the specified buffer size, and saves its argument, the input stream in, for later use. |

| **Method Summary** | |
| --- | --- |
| int | [**available**](http://docs.google.com/java/io/BufferedInputStream.html#available())()            Returns an estimate of the number of bytes that can be read (or skipped over) from this input stream without blocking by the next invocation of a method for this input stream. |
| void | [**close**](http://docs.google.com/java/io/BufferedInputStream.html#close())()            Closes this input stream and releases any system resources associated with the stream. |
| void | [**mark**](http://docs.google.com/java/io/BufferedInputStream.html#mark(int))(int readlimit)            See the general contract of the mark method of InputStream. |
| boolean | [**markSupported**](http://docs.google.com/java/io/BufferedInputStream.html#markSupported())()            Tests if this input stream supports the mark and reset methods. |
| int | [**read**](http://docs.google.com/java/io/BufferedInputStream.html#read())()            See the general contract of the read method of InputStream. |
| int | [**read**](http://docs.google.com/java/io/BufferedInputStream.html#read(byte%5B%5D,%20int,%20int))(byte[] b, int off, int len)            Reads bytes from this byte-input stream into the specified byte array, starting at the given offset. |
| void | [**reset**](http://docs.google.com/java/io/BufferedInputStream.html#reset())()            See the general contract of the reset method of InputStream. |
| long | [**skip**](http://docs.google.com/java/io/BufferedInputStream.html#skip(long))(long n)            See the general contract of the skip method of InputStream. |

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

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

### buf

protected volatile byte[] **buf**

The internal buffer array where the data is stored. When necessary, it may be replaced by another array of a different size.

### count

protected int **count**

The index one greater than the index of the last valid byte in the buffer. This value is always in the range 0 through buf.length; elements buf[0] through buf[count-1] contain buffered input data obtained from the underlying input stream.

### pos

protected int **pos**

The current position in the buffer. This is the index of the next character to be read from the buf array.

This value is always in the range 0 through count. If it is less than count, then buf[pos] is the next byte to be supplied as input; if it is equal to count, then the next read or skip operation will require more bytes to be read from the contained input stream.

**See Also:**[buf](http://docs.google.com/java/io/BufferedInputStream.html#buf)

### markpos

protected int **markpos**

The value of the pos field at the time the last mark method was called.

This value is always in the range -1 through pos. If there is no marked position in the input stream, this field is -1. If there is a marked position in the input stream, then buf[markpos] is the first byte to be supplied as input after a reset operation. If markpos is not -1, then all bytes from positions buf[markpos] through buf[pos-1] must remain in the buffer array (though they may be moved to another place in the buffer array, with suitable adjustments to the values of count, pos, and markpos); they may not be discarded unless and until the difference between pos and markpos exceeds marklimit.

**See Also:**[mark(int)](http://docs.google.com/java/io/BufferedInputStream.html#mark(int)), [pos](http://docs.google.com/java/io/BufferedInputStream.html#pos)

### marklimit

protected int **marklimit**

The maximum read ahead allowed after a call to the mark method before subsequent calls to the reset method fail. Whenever the difference between pos and markpos exceeds marklimit, then the mark may be dropped by setting markpos to -1.

**See Also:**[mark(int)](http://docs.google.com/java/io/BufferedInputStream.html#mark(int)), [reset()](http://docs.google.com/java/io/BufferedInputStream.html#reset())

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

### BufferedInputStream

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

Creates a BufferedInputStream and saves its argument, the input stream in, for later use. An internal buffer array is created and stored in buf.

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

### BufferedInputStream

public **BufferedInputStream**([InputStream](http://docs.google.com/java/io/InputStream.html) in,  
 int size)

Creates a BufferedInputStream with the specified buffer size, and saves its argument, the input stream in, for later use. An internal buffer array of length size is created and stored in buf.

**Parameters:**in - the underlying input stream.size - the buffer size. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if size <= 0.

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

### read

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

See the general contract of the read method of InputStream.

**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 the stream is reached. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if this input stream has been closed by invoking its [close()](http://docs.google.com/java/io/BufferedInputStream.html#close()) method, or an I/O error occurs.**See Also:**[FilterInputStream.in](http://docs.google.com/java/io/FilterInputStream.html#in)

### read

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

Reads bytes from this byte-input stream into the specified byte array, starting at the given offset.

This method implements the general contract of the corresponding [read](http://docs.google.com/java/io/InputStream.html#read(byte%5B%5D,%20int,%20int)) method of the [InputStream](http://docs.google.com/java/io/InputStream.html) class. As an additional convenience, it attempts to read as many bytes as possible by repeatedly invoking the read method of the underlying stream. This iterated read continues until one of the following conditions becomes true:

* The specified number of bytes have been read,
* The read method of the underlying stream returns -1, indicating end-of-file, or
* The available method of the underlying stream returns zero, indicating that further input requests would block.

If the first read on the underlying stream returns -1 to indicate end-of-file then this method returns -1. Otherwise this method returns the number of bytes actually read.

Subclasses of this class are encouraged, but not required, to attempt to read as many bytes as possible in the same fashion.

**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 - destination buffer.off - offset at which to start storing bytes.len - maximum number of bytes to read. **Returns:**the number of bytes read, or -1 if the end of the stream has been reached. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if this input stream has been closed by invoking its [close()](http://docs.google.com/java/io/BufferedInputStream.html#close()) method, or an I/O error occurs.**See Also:**[FilterInputStream.in](http://docs.google.com/java/io/FilterInputStream.html#in)

### skip

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

See the general contract of the skip method of InputStream.

**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 the stream does not support seek, or if this input stream has been closed by invoking its [close()](http://docs.google.com/java/io/BufferedInputStream.html#close()) method, or an I/O error occurs.

### available

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

Returns an estimate of the number of bytes that can be read (or skipped over) from this input stream without blocking by the next invocation of a method for this input stream. The next invocation might be the same thread or another thread. A single read or skip of this many bytes will not block, but may read or skip fewer bytes.

This method returns the sum of the number of bytes remaining to be read in the buffer (count - pos) and the result of calling the [in](http://docs.google.com/java/io/FilterInputStream.html#in).available().

**Overrides:**[available](http://docs.google.com/java/io/FilterInputStream.html#available()) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Returns:**an estimate of the number of bytes that can be read (or skipped over) from this input stream without blocking. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if this input stream has been closed by invoking its [close()](http://docs.google.com/java/io/BufferedInputStream.html#close()) method, or an I/O error occurs.

### mark

public void **mark**(int readlimit)

See the general contract of the mark method of InputStream.

**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:**[reset()](http://docs.google.com/java/io/BufferedInputStream.html#reset())

### reset

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

See the general contract of the reset method of InputStream.

If markpos is -1 (no mark has been set or the mark has been invalidated), an IOException is thrown. Otherwise, pos is set equal to markpos.

**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 this stream has not been marked or, if the mark has been invalidated, or the stream has been closed by invoking its [close()](http://docs.google.com/java/io/BufferedInputStream.html#close()) method, or an I/O error occurs.**See Also:**[mark(int)](http://docs.google.com/java/io/BufferedInputStream.html#mark(int))

### markSupported

public boolean **markSupported**()

Tests if this input stream supports the mark and reset methods. The markSupported method of BufferedInputStream returns true.

**Overrides:**[markSupported](http://docs.google.com/java/io/FilterInputStream.html#markSupported()) in class [FilterInputStream](http://docs.google.com/java/io/FilterInputStream.html) **Returns:**a boolean indicating if this stream type supports the mark and reset methods.**See Also:**[InputStream.mark(int)](http://docs.google.com/java/io/InputStream.html#mark(int)), [InputStream.reset()](http://docs.google.com/java/io/InputStream.html#reset())

### close

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

Closes this input stream and releases any system resources associated with the stream. Once the stream has been closed, further read(), available(), reset(), or skip() invocations will throw an IOException. Closing a previously closed stream has no effect.

**Specified by:**[close](http://docs.google.com/java/io/Closeable.html#close()) in interface [Closeable](http://docs.google.com/java/io/Closeable.html)**Overrides:**[close](http://docs.google.com/java/io/FilterInputStream.html#close()) 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)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BufferedInputStream.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   [**NEXT CLASS**](http://docs.google.com/java/io/BufferedOutputStream.html) | [**FRAMES**](http://docs.google.com/index.html?java/io/BufferedInputStream.html)    [**NO FRAMES**](http://docs.google.com/BufferedInputStream.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: [FIELD](#2s8eyo1) | [CONSTR](#1ksv4uv) | [METHOD](#z337ya) |

[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).