| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MemoryCacheImageOutputStream.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/javax/imageio/stream/MemoryCacheImageInputStream.html)   NEXT CLASS | [**FRAMES**](http://docs.google.com/index.html?javax/imageio/stream/MemoryCacheImageOutputStream.html)    [**NO FRAMES**](http://docs.google.com/MemoryCacheImageOutputStream.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#3rdcrjn) | [METHOD](#lnxbz9) |

## **javax.imageio.stream**

Class MemoryCacheImageOutputStream

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.imageio.stream.ImageInputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html)  
 [javax.imageio.stream.ImageOutputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html)  
 **javax.imageio.stream.MemoryCacheImageOutputStream**

**All Implemented Interfaces:** [DataInput](http://docs.google.com/java/io/DataInput.html), [DataOutput](http://docs.google.com/java/io/DataOutput.html), [ImageInputStream](http://docs.google.com/javax/imageio/stream/ImageInputStream.html), [ImageOutputStream](http://docs.google.com/javax/imageio/stream/ImageOutputStream.html)

public class **MemoryCacheImageOutputStream**extends [ImageOutputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html)

An implementation of ImageOutputStream that writes its output to a regular OutputStream. A memory buffer is used to cache at least the data between the discard position and the current write position. The only constructor takes an OutputStream, so this class may not be used for read/modify/write operations. Reading can occur only on parts of the stream that have already been written to the cache and not yet flushed.

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

| **Fields inherited from class javax.imageio.stream.**[**ImageInputStreamImpl**](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html) |
| --- |
| [bitOffset](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#bitOffset), [byteOrder](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#byteOrder), [flushedPos](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#flushedPos), [streamPos](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#streamPos) |

| **Constructor Summary** | |
| --- | --- |
| [**MemoryCacheImageOutputStream**](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#MemoryCacheImageOutputStream(java.io.OutputStream))([OutputStream](http://docs.google.com/java/io/OutputStream.html) stream)            Constructs a MemoryCacheImageOutputStream that will write to a given OutputStream. |

| **Method Summary** | |
| --- | --- |
| void | [**close**](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#close())()            Closes this MemoryCacheImageOutputStream. |
| void | [**flushBefore**](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#flushBefore(long))(long pos)            Discards the initial portion of the stream prior to the indicated postion. |
| boolean | [**isCached**](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#isCached())()            Returns true since this ImageOutputStream caches data in order to allow seeking backwards. |
| boolean | [**isCachedFile**](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#isCachedFile())()            Returns false since this ImageOutputStream does not maintain a file cache. |
| boolean | [**isCachedMemory**](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#isCachedMemory())()            Returns true since this ImageOutputStream maintains a main memory cache. |
| long | [**length**](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#length())()            Returns -1L to indicate that the stream has unknown length. |
| int | [**read**](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#read())()            Reads a single byte from the stream and returns it as an int between 0 and 255. |
| int | [**read**](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#read(byte%5B%5D,%20int,%20int))(byte[] b, int off, int len)            Reads up to len bytes from the stream, and stores them into b starting at index off. |
| void | [**write**](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#write(byte%5B%5D,%20int,%20int))(byte[] b, int off, int len)            Writes a sequence of bytes to the stream at the current position. |
| void | [**write**](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#write(int))(int b)            Writes a single byte to the stream at the current position. |

| **Methods inherited from class javax.imageio.stream.**[**ImageOutputStreamImpl**](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html) |
| --- |
| [flushBits](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#flushBits()), [write](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#write(byte%5B%5D)), [writeBit](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeBit(int)), [writeBits](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeBits(long,%20int)), [writeBoolean](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeBoolean(boolean)), [writeByte](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeByte(int)), [writeBytes](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeBytes(java.lang.String)), [writeChar](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeChar(int)), [writeChars](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeChars(char%5B%5D,%20int,%20int)), [writeChars](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeChars(java.lang.String)), [writeDouble](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeDouble(double)), [writeDoubles](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeDoubles(double%5B%5D,%20int,%20int)), [writeFloat](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeFloat(float)), [writeFloats](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeFloats(float%5B%5D,%20int,%20int)), [writeInt](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeInt(int)), [writeInts](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeInts(int%5B%5D,%20int,%20int)), [writeLong](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeLong(long)), [writeLongs](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeLongs(long%5B%5D,%20int,%20int)), [writeShort](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeShort(int)), [writeShorts](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeShorts(short%5B%5D,%20int,%20int)), [writeUTF](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#writeUTF(java.lang.String)) |

| **Methods inherited from class javax.imageio.stream.**[**ImageInputStreamImpl**](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html) |
| --- |
| [checkClosed](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#checkClosed()), [finalize](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#finalize()), [flush](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#flush()), [getBitOffset](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#getBitOffset()), [getByteOrder](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#getByteOrder()), [getFlushedPosition](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#getFlushedPosition()), [getStreamPosition](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#getStreamPosition()), [mark](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#mark()), [read](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#read(byte%5B%5D)), [readBit](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readBit()), [readBits](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readBits(int)), [readBoolean](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readBoolean()), [readByte](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readByte()), [readBytes](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readBytes(javax.imageio.stream.IIOByteBuffer,%20int)), [readChar](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readChar()), [readDouble](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readDouble()), [readFloat](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readFloat()), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readFully(byte%5B%5D)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readFully(byte%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readFully(char%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readFully(double%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readFully(float%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readFully(int%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readFully(long%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readFully(short%5B%5D,%20int,%20int)), [readInt](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readInt()), [readLine](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readLine()), [readLong](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readLong()), [readShort](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readShort()), [readUnsignedByte](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readUnsignedByte()), [readUnsignedInt](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readUnsignedInt()), [readUnsignedShort](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readUnsignedShort()), [readUTF](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#readUTF()), [reset](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#reset()), [seek](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#seek(long)), [setBitOffset](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#setBitOffset(int)), [setByteOrder](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#setByteOrder(java.nio.ByteOrder)), [skipBytes](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#skipBytes(int)), [skipBytes](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#skipBytes(long)) |

| **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)) |

| **Methods inherited from interface javax.imageio.stream.**[**ImageInputStream**](http://docs.google.com/javax/imageio/stream/ImageInputStream.html) |
| --- |
| [flush](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#flush()), [getBitOffset](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#getBitOffset()), [getByteOrder](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#getByteOrder()), [getFlushedPosition](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#getFlushedPosition()), [getStreamPosition](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#getStreamPosition()), [mark](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#mark()), [read](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#read(byte%5B%5D)), [readBit](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readBit()), [readBits](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readBits(int)), [readBoolean](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readBoolean()), [readByte](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readByte()), [readBytes](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readBytes(javax.imageio.stream.IIOByteBuffer,%20int)), [readChar](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readChar()), [readDouble](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readDouble()), [readFloat](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readFloat()), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readFully(byte%5B%5D)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readFully(byte%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readFully(char%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readFully(double%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readFully(float%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readFully(int%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readFully(long%5B%5D,%20int,%20int)), [readFully](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readFully(short%5B%5D,%20int,%20int)), [readInt](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readInt()), [readLine](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readLine()), [readLong](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readLong()), [readShort](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readShort()), [readUnsignedByte](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readUnsignedByte()), [readUnsignedInt](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readUnsignedInt()), [readUnsignedShort](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readUnsignedShort()), [readUTF](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#readUTF()), [reset](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#reset()), [seek](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#seek(long)), [setBitOffset](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#setBitOffset(int)), [setByteOrder](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#setByteOrder(java.nio.ByteOrder)), [skipBytes](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#skipBytes(int)), [skipBytes](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#skipBytes(long)) |

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

### MemoryCacheImageOutputStream

public **MemoryCacheImageOutputStream**([OutputStream](http://docs.google.com/java/io/OutputStream.html) stream)

Constructs a MemoryCacheImageOutputStream that will write to a given OutputStream.

**Parameters:**stream - an OutputStream to write to. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if stream is null.

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

### read

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

**Description copied from class:** [**ImageInputStreamImpl**](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#read()) Reads a single byte from the stream and returns it as an int between 0 and 255. If EOF is reached, -1 is returned.

Subclasses must provide an implementation for this method. The subclass implementation should update the stream position before exiting.

The bit offset within the stream must be reset to zero before the read occurs.

**Specified by:**[read](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#read()) in interface [ImageInputStream](http://docs.google.com/javax/imageio/stream/ImageInputStream.html)**Specified by:**[read](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#read()) in class [ImageInputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html) **Returns:**the value of the next byte in the stream, or -1 if EOF is reached. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if the stream has been closed.

### read

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

**Description copied from class:** [**ImageInputStreamImpl**](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#read(byte%5B%5D,%20int,%20int)) Reads up to len bytes from the stream, and stores them into b starting at index off. If no bytes can be read because the end of the stream has been reached, -1 is returned.

The bit offset within the stream must be reset to zero before the read occurs.

Subclasses must provide an implementation for this method. The subclass implementation should update the stream position before exiting.

**Specified by:**[read](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#read(byte%5B%5D,%20int,%20int)) in interface [ImageInputStream](http://docs.google.com/javax/imageio/stream/ImageInputStream.html)**Specified by:**[read](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#read(byte%5B%5D,%20int,%20int)) in class [ImageInputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html) **Parameters:**b - an array of bytes to be written to.off - the starting position within b to write to.len - the maximum number of bytes to read. **Returns:**the number of bytes actually read, or -1 to indicate EOF. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### write

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

**Description copied from interface:** [**ImageOutputStream**](http://docs.google.com/javax/imageio/stream/ImageOutputStream.html#write(int)) Writes a single byte to the stream at the current position. The 24 high-order bits of b are ignored.

If the bit offset within the stream is non-zero, the remainder of the current byte is padded with 0s and written out first. The bit offset will be 0 after the write. Implementers can use the [flushBits](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#flushBits()) method of [ImageOutputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html) to guarantee this.

**Specified by:**[write](http://docs.google.com/java/io/DataOutput.html#write(int)) in interface [DataOutput](http://docs.google.com/java/io/DataOutput.html)**Specified by:**[write](http://docs.google.com/javax/imageio/stream/ImageOutputStream.html#write(int)) in interface [ImageOutputStream](http://docs.google.com/javax/imageio/stream/ImageOutputStream.html)**Specified by:**[write](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#write(int)) in class [ImageOutputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html) **Parameters:**b - an int whose lower 8 bits are to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### write

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

**Description copied from interface:** [**ImageOutputStream**](http://docs.google.com/javax/imageio/stream/ImageOutputStream.html#write(byte%5B%5D,%20int,%20int)) Writes a sequence of bytes to the stream at the current position. If len is 0, nothing is written. The byte b[off] is written first, then the byte b[off + 1], and so on.

If the bit offset within the stream is non-zero, the remainder of the current byte is padded with 0s and written out first. The bit offset will be 0 after the write. Implementers can use the [flushBits](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#flushBits()) method of [ImageOutputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html) to guarantee this.

**Specified by:**[write](http://docs.google.com/java/io/DataOutput.html#write(byte%5B%5D,%20int,%20int)) in interface [DataOutput](http://docs.google.com/java/io/DataOutput.html)**Specified by:**[write](http://docs.google.com/javax/imageio/stream/ImageOutputStream.html#write(byte%5B%5D,%20int,%20int)) in interface [ImageOutputStream](http://docs.google.com/javax/imageio/stream/ImageOutputStream.html)**Specified by:**[write](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html#write(byte%5B%5D,%20int,%20int)) in class [ImageOutputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html) **Parameters:**b - an array of bytes to be written.off - the start offset in the data.len - the number of bytes to write. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### length

public long **length**()

**Description copied from class:** [**ImageInputStreamImpl**](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#length()) Returns -1L to indicate that the stream has unknown length. Subclasses must override this method to provide actual length information.

**Specified by:**[length](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#length()) in interface [ImageInputStream](http://docs.google.com/javax/imageio/stream/ImageInputStream.html)**Overrides:**[length](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#length()) in class [ImageInputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html) **Returns:**-1L to indicate unknown length.

### isCached

public boolean **isCached**()

Returns true since this ImageOutputStream caches data in order to allow seeking backwards.

**Specified by:**[isCached](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#isCached()) in interface [ImageInputStream](http://docs.google.com/javax/imageio/stream/ImageInputStream.html)**Overrides:**[isCached](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#isCached()) in class [ImageInputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html) **Returns:**true.**See Also:**[isCachedMemory()](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#isCachedMemory()), [isCachedFile()](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#isCachedFile())

### isCachedFile

public boolean **isCachedFile**()

Returns false since this ImageOutputStream does not maintain a file cache.

**Specified by:**[isCachedFile](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#isCachedFile()) in interface [ImageInputStream](http://docs.google.com/javax/imageio/stream/ImageInputStream.html)**Overrides:**[isCachedFile](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#isCachedFile()) in class [ImageInputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html) **Returns:**false.**See Also:**[isCached()](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#isCached()), [isCachedMemory()](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#isCachedMemory())

### isCachedMemory

public boolean **isCachedMemory**()

Returns true since this ImageOutputStream maintains a main memory cache.

**Specified by:**[isCachedMemory](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#isCachedMemory()) in interface [ImageInputStream](http://docs.google.com/javax/imageio/stream/ImageInputStream.html)**Overrides:**[isCachedMemory](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#isCachedMemory()) in class [ImageInputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html) **Returns:**true.**See Also:**[isCached()](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#isCached()), [isCachedFile()](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html#isCachedFile())

### close

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

Closes this MemoryCacheImageOutputStream. All pending data is flushed to the output, and the cache is released. The destination OutputStream is not closed.

**Specified by:**[close](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#close()) in interface [ImageInputStream](http://docs.google.com/javax/imageio/stream/ImageInputStream.html)**Overrides:**[close](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#close()) in class [ImageInputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html) **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### flushBefore

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

**Description copied from interface:** [**ImageInputStream**](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#flushBefore(long)) Discards the initial portion of the stream prior to the indicated postion. Attempting to seek to an offset within the flushed portion of the stream will result in an IndexOutOfBoundsException.

Calling flushBefore may allow classes implementing this interface to free up resources such as memory or disk space that are being used to store data from the stream.

**Specified by:**[flushBefore](http://docs.google.com/javax/imageio/stream/ImageInputStream.html#flushBefore(long)) in interface [ImageInputStream](http://docs.google.com/javax/imageio/stream/ImageInputStream.html)**Specified by:**[flushBefore](http://docs.google.com/javax/imageio/stream/ImageOutputStream.html#flushBefore(long)) in interface [ImageOutputStream](http://docs.google.com/javax/imageio/stream/ImageOutputStream.html)**Overrides:**[flushBefore](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#flushBefore(long)) in class [ImageInputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html) **Parameters:**pos - a long containing the length of the stream prefix that may be flushed. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MemoryCacheImageOutputStream.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/javax/imageio/stream/MemoryCacheImageInputStream.html)   NEXT CLASS | [**FRAMES**](http://docs.google.com/index.html?javax/imageio/stream/MemoryCacheImageOutputStream.html)    [**NO FRAMES**](http://docs.google.com/MemoryCacheImageOutputStream.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#3rdcrjn) | [METHOD](#lnxbz9) |

[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).