| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/FileCacheImageInputStream.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/javax/imageio/stream/FileCacheImageOutputStream.html) | [**FRAMES**](http://docs.google.com/index.html?javax/imageio/stream/FileCacheImageInputStream.html)    [**NO FRAMES**](http://docs.google.com/FileCacheImageInputStream.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#3rdcrjn) |

## **javax.imageio.stream**

Class FileCacheImageInputStream

[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.FileCacheImageInputStream**

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

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

An implementation of ImageInputStream that gets its input from a regular InputStream. A file is used to cache previously read data.

| **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** | |
| --- | --- |
| [**FileCacheImageInputStream**](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.html#FileCacheImageInputStream(java.io.InputStream,%20java.io.File))([InputStream](http://docs.google.com/java/io/InputStream.html) stream, [File](http://docs.google.com/java/io/File.html) cacheDir)            Constructs a FileCacheImageInputStream that will read from a given InputStream. |

| **Method Summary** | |
| --- | --- |
| void | [**close**](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.html#close())()            Closes this FileCacheImageInputStream, closing and removing the cache file. |
| protected  void | [**finalize**](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.html#finalize())()            Finalizes this object prior to garbage collection. |
| boolean | [**isCached**](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.html#isCached())()            Returns true since this ImageInputStream caches data in order to allow seeking backwards. |
| boolean | [**isCachedFile**](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.html#isCachedFile())()            Returns true since this ImageInputStream maintains a file cache. |
| boolean | [**isCachedMemory**](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.html#isCachedMemory())()            Returns false since this ImageInputStream does not maintain a main memory cache. |
| int | [**read**](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.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/FileCacheImageInputStream.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. |

| **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()), [flush](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#flush()), [flushBefore](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#flushBefore(long)), [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()), [length](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#length()), [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)) |

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

### FileCacheImageInputStream

public **FileCacheImageInputStream**([InputStream](http://docs.google.com/java/io/InputStream.html) stream,  
 [File](http://docs.google.com/java/io/File.html) cacheDir)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Constructs a FileCacheImageInputStream that will read from a given InputStream.

A temporary file is used as a cache. If cacheDiris non-null and is a directory, the file will be created there. If it is null, the system-dependent default temporary-file directory will be used (see the documentation for File.createTempFile for details).

**Parameters:**stream - an InputStream to read from.cacheDir - a File indicating where the cache file should be created, or null to use the system directory. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if stream is null. [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if cacheDir is non-null but is not a directory. [IOException](http://docs.google.com/java/io/IOException.html) - if a cache file cannot be created.

| **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.

### isCached

public boolean **isCached**()

Returns true since this ImageInputStream 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/FileCacheImageInputStream.html#isCachedMemory()), [isCachedFile()](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.html#isCachedFile())

### isCachedFile

public boolean **isCachedFile**()

Returns true since this ImageInputStream maintains 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:**true.**See Also:**[isCached()](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.html#isCached()), [isCachedMemory()](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.html#isCachedMemory())

### isCachedMemory

public boolean **isCachedMemory**()

Returns false since this ImageInputStream does not maintain 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:**false.**See Also:**[isCached()](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.html#isCached()), [isCachedFile()](http://docs.google.com/javax/imageio/stream/FileCacheImageInputStream.html#isCachedFile())

### close

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

Closes this FileCacheImageInputStream, closing and removing the cache file. The source InputStream 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 error occurs.

### finalize

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

Finalizes this object prior to garbage collection. The close method is called to close any open input source. This method should not be called from application code.

**Overrides:**[finalize](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html#finalize()) in class [ImageInputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageInputStreamImpl.html) **Throws:** [Throwable](http://docs.google.com/java/lang/Throwable.html) - if an error occurs during superclass finalization.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/FileCacheImageInputStream.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/javax/imageio/stream/FileCacheImageOutputStream.html) | [**FRAMES**](http://docs.google.com/index.html?javax/imageio/stream/FileCacheImageInputStream.html)    [**NO FRAMES**](http://docs.google.com/FileCacheImageInputStream.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).