| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/JPEGImageReadParam.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/plugins/jpeg/JPEGHuffmanTable.html)   [**NEXT CLASS**](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageWriteParam.html) | [**FRAMES**](http://docs.google.com/index.html?javax/imageio/plugins/jpeg/JPEGImageReadParam.html)    [**NO FRAMES**](http://docs.google.com/JPEGImageReadParam.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: FIELD | [CONSTR](#3rdcrjn) | [METHOD](#lnxbz9) |

## **javax.imageio.plugins.jpeg**

Class JPEGImageReadParam

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.imageio.IIOParam](http://docs.google.com/javax/imageio/IIOParam.html)  
 [javax.imageio.ImageReadParam](http://docs.google.com/javax/imageio/ImageReadParam.html)  
 **javax.imageio.plugins.jpeg.JPEGImageReadParam**

public class **JPEGImageReadParam**extends [ImageReadParam](http://docs.google.com/javax/imageio/ImageReadParam.html)

This class adds the ability to set JPEG quantization and Huffman tables when using the built-in JPEG reader plug-in. An instance of this class will be returned from the getDefaultImageReadParam methods of the built-in JPEG ImageReader.

The sole purpose of these additions is to allow the specification of tables for use in decoding abbreviated streams. The built-in JPEG reader will also accept an ordinary ImageReadParam, which is sufficient for decoding non-abbreviated streams.

While tables for abbreviated streams are often obtained by first reading another abbreviated stream containing only the tables, in some applications the tables are fixed ahead of time. This class allows the tables to be specified directly from client code. If no tables are specified either in the stream or in a JPEGImageReadParam, then the stream is presumed to use the "standard" visually lossless tables. See [JPEGQTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGQTable.html) and [JPEGHuffmanTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGHuffmanTable.html) for more information on the default tables.

The default JPEGImageReadParam returned by the getDefaultReadParam method of the builtin JPEG reader contains no tables. Default tables may be obtained from the table classes [JPEGQTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGQTable.html) and [JPEGHuffmanTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGHuffmanTable.html).

If a stream does contain tables, the tables given in a JPEGImageReadParam are ignored. Furthermore, if the first image in a stream does contain tables and subsequent ones do not, then the tables given in the first image are used for all the abbreviated images. Once tables have been read from a stream, they can be overridden only by tables subsequently read from the same stream. In order to specify new tables, the [setInput](http://docs.google.com/javax/imageio/ImageReader.html#setInput(java.lang.Object,%20boolean,%20boolean)) method of the reader must be called to change the stream.

Note that this class does not provide a means for obtaining the tables found in a stream. These may be extracted from a stream by consulting the IIOMetadata object returned by the reader.

For more information about the operation of the built-in JPEG plug-ins, see the [JPEG metadata format specification and usage notes](http://docs.google.com/metadata/doc-files/jpeg_metadata.html).

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

| **Fields inherited from class javax.imageio.**[**ImageReadParam**](http://docs.google.com/javax/imageio/ImageReadParam.html) |
| --- |
| [canSetSourceRenderSize](http://docs.google.com/javax/imageio/ImageReadParam.html#canSetSourceRenderSize), [destination](http://docs.google.com/javax/imageio/ImageReadParam.html#destination), [destinationBands](http://docs.google.com/javax/imageio/ImageReadParam.html#destinationBands), [minProgressivePass](http://docs.google.com/javax/imageio/ImageReadParam.html#minProgressivePass), [numProgressivePasses](http://docs.google.com/javax/imageio/ImageReadParam.html#numProgressivePasses), [sourceRenderSize](http://docs.google.com/javax/imageio/ImageReadParam.html#sourceRenderSize) |

| **Fields inherited from class javax.imageio.**[**IIOParam**](http://docs.google.com/javax/imageio/IIOParam.html) |
| --- |
| [controller](http://docs.google.com/javax/imageio/IIOParam.html#controller), [defaultController](http://docs.google.com/javax/imageio/IIOParam.html#defaultController), [destinationOffset](http://docs.google.com/javax/imageio/IIOParam.html#destinationOffset), [destinationType](http://docs.google.com/javax/imageio/IIOParam.html#destinationType), [sourceBands](http://docs.google.com/javax/imageio/IIOParam.html#sourceBands), [sourceRegion](http://docs.google.com/javax/imageio/IIOParam.html#sourceRegion), [sourceXSubsampling](http://docs.google.com/javax/imageio/IIOParam.html#sourceXSubsampling), [sourceYSubsampling](http://docs.google.com/javax/imageio/IIOParam.html#sourceYSubsampling), [subsamplingXOffset](http://docs.google.com/javax/imageio/IIOParam.html#subsamplingXOffset), [subsamplingYOffset](http://docs.google.com/javax/imageio/IIOParam.html#subsamplingYOffset) |

| **Constructor Summary** | |
| --- | --- |
| [**JPEGImageReadParam**](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#JPEGImageReadParam())()            Constructs a JPEGImageReadParam. |

| **Method Summary** | |
| --- | --- |
| boolean | [**areTablesSet**](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#areTablesSet())()            Returns true if tables are currently set. |
| [JPEGHuffmanTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGHuffmanTable.html)[] | [**getACHuffmanTables**](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#getACHuffmanTables())()            Returns a copy of the array of AC Huffman tables set on the most recent call to setDecodeTables, or null if tables are not currently set. |
| [JPEGHuffmanTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGHuffmanTable.html)[] | [**getDCHuffmanTables**](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#getDCHuffmanTables())()            Returns a copy of the array of DC Huffman tables set on the most recent call to setDecodeTables, or null if tables are not currently set. |
| [JPEGQTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGQTable.html)[] | [**getQTables**](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#getQTables())()            Returns a copy of the array of quantization tables set on the most recent call to setDecodeTables, or null if tables are not currently set. |
| void | [**setDecodeTables**](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#setDecodeTables(javax.imageio.plugins.jpeg.JPEGQTable%5B%5D,%20javax.imageio.plugins.jpeg.JPEGHuffmanTable%5B%5D,%20javax.imageio.plugins.jpeg.JPEGHuffmanTable%5B%5D))([JPEGQTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGQTable.html)[] qTables, [JPEGHuffmanTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGHuffmanTable.html)[] DCHuffmanTables, [JPEGHuffmanTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGHuffmanTable.html)[] ACHuffmanTables)            Sets the quantization and Huffman tables to use in decoding abbreviated streams. |
| void | [**unsetDecodeTables**](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#unsetDecodeTables())()            Removes any quantization and Huffman tables that are currently set. |

| **Methods inherited from class javax.imageio.**[**ImageReadParam**](http://docs.google.com/javax/imageio/ImageReadParam.html) |
| --- |
| [canSetSourceRenderSize](http://docs.google.com/javax/imageio/ImageReadParam.html#canSetSourceRenderSize()), [getDestination](http://docs.google.com/javax/imageio/ImageReadParam.html#getDestination()), [getDestinationBands](http://docs.google.com/javax/imageio/ImageReadParam.html#getDestinationBands()), [getSourceMaxProgressivePass](http://docs.google.com/javax/imageio/ImageReadParam.html#getSourceMaxProgressivePass()), [getSourceMinProgressivePass](http://docs.google.com/javax/imageio/ImageReadParam.html#getSourceMinProgressivePass()), [getSourceNumProgressivePasses](http://docs.google.com/javax/imageio/ImageReadParam.html#getSourceNumProgressivePasses()), [getSourceRenderSize](http://docs.google.com/javax/imageio/ImageReadParam.html#getSourceRenderSize()), [setDestination](http://docs.google.com/javax/imageio/ImageReadParam.html#setDestination(java.awt.image.BufferedImage)), [setDestinationBands](http://docs.google.com/javax/imageio/ImageReadParam.html#setDestinationBands(int%5B%5D)), [setDestinationType](http://docs.google.com/javax/imageio/ImageReadParam.html#setDestinationType(javax.imageio.ImageTypeSpecifier)), [setSourceProgressivePasses](http://docs.google.com/javax/imageio/ImageReadParam.html#setSourceProgressivePasses(int,%20int)), [setSourceRenderSize](http://docs.google.com/javax/imageio/ImageReadParam.html#setSourceRenderSize(java.awt.Dimension)) |

| **Methods inherited from class javax.imageio.**[**IIOParam**](http://docs.google.com/javax/imageio/IIOParam.html) |
| --- |
| [activateController](http://docs.google.com/javax/imageio/IIOParam.html#activateController()), [getController](http://docs.google.com/javax/imageio/IIOParam.html#getController()), [getDefaultController](http://docs.google.com/javax/imageio/IIOParam.html#getDefaultController()), [getDestinationOffset](http://docs.google.com/javax/imageio/IIOParam.html#getDestinationOffset()), [getDestinationType](http://docs.google.com/javax/imageio/IIOParam.html#getDestinationType()), [getSourceBands](http://docs.google.com/javax/imageio/IIOParam.html#getSourceBands()), [getSourceRegion](http://docs.google.com/javax/imageio/IIOParam.html#getSourceRegion()), [getSourceXSubsampling](http://docs.google.com/javax/imageio/IIOParam.html#getSourceXSubsampling()), [getSourceYSubsampling](http://docs.google.com/javax/imageio/IIOParam.html#getSourceYSubsampling()), [getSubsamplingXOffset](http://docs.google.com/javax/imageio/IIOParam.html#getSubsamplingXOffset()), [getSubsamplingYOffset](http://docs.google.com/javax/imageio/IIOParam.html#getSubsamplingYOffset()), [hasController](http://docs.google.com/javax/imageio/IIOParam.html#hasController()), [setController](http://docs.google.com/javax/imageio/IIOParam.html#setController(javax.imageio.IIOParamController)), [setDestinationOffset](http://docs.google.com/javax/imageio/IIOParam.html#setDestinationOffset(java.awt.Point)), [setSourceBands](http://docs.google.com/javax/imageio/IIOParam.html#setSourceBands(int%5B%5D)), [setSourceRegion](http://docs.google.com/javax/imageio/IIOParam.html#setSourceRegion(java.awt.Rectangle)), [setSourceSubsampling](http://docs.google.com/javax/imageio/IIOParam.html#setSourceSubsampling(int,%20int,%20int,%20int)) |

| **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** |
| --- |

### JPEGImageReadParam

public **JPEGImageReadParam**()

Constructs a JPEGImageReadParam.

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

### areTablesSet

public boolean **areTablesSet**()

Returns true if tables are currently set.

**Returns:**true if tables are present.

### setDecodeTables

public void **setDecodeTables**([JPEGQTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGQTable.html)[] qTables,  
 [JPEGHuffmanTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGHuffmanTable.html)[] DCHuffmanTables,  
 [JPEGHuffmanTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGHuffmanTable.html)[] ACHuffmanTables)

Sets the quantization and Huffman tables to use in decoding abbreviated streams. There may be a maximum of 4 tables of each type. These tables are ignored once tables are encountered in the stream. All arguments must be non-null. The two arrays of Huffman tables must have the same number of elements. The table specifiers in the frame and scan headers in the stream are assumed to be equivalent to indices into these arrays. The argument arrays are copied by this method.

**Parameters:**qTables - an array of quantization table objects.DCHuffmanTables - an array of Huffman table objects.ACHuffmanTables - an array of Huffman table objects. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if any of the arguments is null, has more than 4 elements, or if the numbers of DC and AC tables differ.**See Also:**[unsetDecodeTables()](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#unsetDecodeTables())

### unsetDecodeTables

public void **unsetDecodeTables**()

Removes any quantization and Huffman tables that are currently set.

**See Also:**[setDecodeTables(javax.imageio.plugins.jpeg.JPEGQTable[], javax.imageio.plugins.jpeg.JPEGHuffmanTable[], javax.imageio.plugins.jpeg.JPEGHuffmanTable[])](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#setDecodeTables(javax.imageio.plugins.jpeg.JPEGQTable%5B%5D,%20javax.imageio.plugins.jpeg.JPEGHuffmanTable%5B%5D,%20javax.imageio.plugins.jpeg.JPEGHuffmanTable%5B%5D))

### getQTables

public [JPEGQTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGQTable.html)[] **getQTables**()

Returns a copy of the array of quantization tables set on the most recent call to setDecodeTables, or null if tables are not currently set.

**Returns:**an array of JPEGQTable objects, or null.**See Also:**[setDecodeTables(javax.imageio.plugins.jpeg.JPEGQTable[], javax.imageio.plugins.jpeg.JPEGHuffmanTable[], javax.imageio.plugins.jpeg.JPEGHuffmanTable[])](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#setDecodeTables(javax.imageio.plugins.jpeg.JPEGQTable%5B%5D,%20javax.imageio.plugins.jpeg.JPEGHuffmanTable%5B%5D,%20javax.imageio.plugins.jpeg.JPEGHuffmanTable%5B%5D))

### getDCHuffmanTables

public [JPEGHuffmanTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGHuffmanTable.html)[] **getDCHuffmanTables**()

Returns a copy of the array of DC Huffman tables set on the most recent call to setDecodeTables, or null if tables are not currently set.

**Returns:**an array of JPEGHuffmanTable objects, or null.**See Also:**[setDecodeTables(javax.imageio.plugins.jpeg.JPEGQTable[], javax.imageio.plugins.jpeg.JPEGHuffmanTable[], javax.imageio.plugins.jpeg.JPEGHuffmanTable[])](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#setDecodeTables(javax.imageio.plugins.jpeg.JPEGQTable%5B%5D,%20javax.imageio.plugins.jpeg.JPEGHuffmanTable%5B%5D,%20javax.imageio.plugins.jpeg.JPEGHuffmanTable%5B%5D))

### getACHuffmanTables

public [JPEGHuffmanTable](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGHuffmanTable.html)[] **getACHuffmanTables**()

Returns a copy of the array of AC Huffman tables set on the most recent call to setDecodeTables, or null if tables are not currently set.

**Returns:**an array of JPEGHuffmanTable objects, or null.**See Also:**[setDecodeTables(javax.imageio.plugins.jpeg.JPEGQTable[], javax.imageio.plugins.jpeg.JPEGHuffmanTable[], javax.imageio.plugins.jpeg.JPEGHuffmanTable[])](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageReadParam.html#setDecodeTables(javax.imageio.plugins.jpeg.JPEGQTable%5B%5D,%20javax.imageio.plugins.jpeg.JPEGHuffmanTable%5B%5D,%20javax.imageio.plugins.jpeg.JPEGHuffmanTable%5B%5D))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/JPEGImageReadParam.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/plugins/jpeg/JPEGHuffmanTable.html)   [**NEXT CLASS**](http://docs.google.com/javax/imageio/plugins/jpeg/JPEGImageWriteParam.html) | [**FRAMES**](http://docs.google.com/index.html?javax/imageio/plugins/jpeg/JPEGImageReadParam.html)    [**NO FRAMES**](http://docs.google.com/JPEGImageReadParam.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | 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).