| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.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 PACKAGE**](http://docs.google.com/java/awt/im/spi/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/java/awt/image/renderable/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

## Package java.awt.image

Provides classes for creating and modifying images.

**See:**

[**Description**](#3znysh7)

| **Interface Summary** | |
| --- | --- |
| [**BufferedImageOp**](http://docs.google.com/java/awt/image/BufferedImageOp.html) | This interface describes single-input/single-output operations performed on BufferedImage objects. |
| [**ImageConsumer**](http://docs.google.com/java/awt/image/ImageConsumer.html) | The interface for objects expressing interest in image data through the ImageProducer interfaces. |
| [**ImageObserver**](http://docs.google.com/java/awt/image/ImageObserver.html) | An asynchronous update interface for receiving notifications about Image information as the Image is constructed. |
| [**ImageProducer**](http://docs.google.com/java/awt/image/ImageProducer.html) | The interface for objects which can produce the image data for Images. |
| [**RasterOp**](http://docs.google.com/java/awt/image/RasterOp.html) | This interface describes single-input/single-output operations performed on Raster objects. |
| [**RenderedImage**](http://docs.google.com/java/awt/image/RenderedImage.html) | RenderedImage is a common interface for objects which contain or can produce image data in the form of Rasters. |
| [**TileObserver**](http://docs.google.com/java/awt/image/TileObserver.html) | An interface for objects that wish to be informed when tiles of a WritableRenderedImage become modifiable by some writer via a call to getWritableTile, and when they become unmodifiable via the last call to releaseWritableTile. |
| [**WritableRenderedImage**](http://docs.google.com/java/awt/image/WritableRenderedImage.html) | WriteableRenderedImage is a common interface for objects which contain or can produce image data in the form of Rasters and which can be modified and/or written over. |

| **Class Summary** | |
| --- | --- |
| [**AffineTransformOp**](http://docs.google.com/java/awt/image/AffineTransformOp.html) | This class uses an affine transform to perform a linear mapping from 2D coordinates in the source image or Raster to 2D coordinates in the destination image or Raster. |
| [**AreaAveragingScaleFilter**](http://docs.google.com/java/awt/image/AreaAveragingScaleFilter.html) | An ImageFilter class for scaling images using a simple area averaging algorithm that produces smoother results than the nearest neighbor algorithm. |
| [**BandCombineOp**](http://docs.google.com/java/awt/image/BandCombineOp.html) | This class performs an arbitrary linear combination of the bands in a Raster, using a specified matrix. |
| [**BandedSampleModel**](http://docs.google.com/java/awt/image/BandedSampleModel.html) | This class represents image data which is stored in a band interleaved fashion and for which each sample of a pixel occupies one data element of the DataBuffer. |
| [**BufferedImage**](http://docs.google.com/java/awt/image/BufferedImage.html) | The BufferedImage subclass describes an [Image](http://docs.google.com/java/awt/Image.html) with an accessible buffer of image data. |
| [**BufferedImageFilter**](http://docs.google.com/java/awt/image/BufferedImageFilter.html) | The BufferedImageFilter class subclasses an ImageFilter to provide a simple means of using a single-source/single-destination image operator ([BufferedImageOp](http://docs.google.com/java/awt/image/BufferedImageOp.html)) to filter a BufferedImage in the Image Producer/Consumer/Observer paradigm. |
| [**BufferStrategy**](http://docs.google.com/java/awt/image/BufferStrategy.html) | The BufferStrategy class represents the mechanism with which to organize complex memory on a particular Canvas or Window. |
| [**ByteLookupTable**](http://docs.google.com/java/awt/image/ByteLookupTable.html) | This class defines a lookup table object. |
| [**ColorConvertOp**](http://docs.google.com/java/awt/image/ColorConvertOp.html) | This class performs a pixel-by-pixel color conversion of the data in the source image. |
| [**ColorModel**](http://docs.google.com/java/awt/image/ColorModel.html) | The ColorModel abstract class encapsulates the methods for translating a pixel value to color components (for example, red, green, and blue) and an alpha component. |
| [**ComponentColorModel**](http://docs.google.com/java/awt/image/ComponentColorModel.html) | A ColorModel class that works with pixel values that represent color and alpha information as separate samples and that store each sample in a separate data element. |
| [**ComponentSampleModel**](http://docs.google.com/java/awt/image/ComponentSampleModel.html) | This class represents image data which is stored such that each sample of a pixel occupies one data element of the DataBuffer. |
| [**ConvolveOp**](http://docs.google.com/java/awt/image/ConvolveOp.html) | This class implements a convolution from the source to the destination. |
| [**CropImageFilter**](http://docs.google.com/java/awt/image/CropImageFilter.html) | An ImageFilter class for cropping images. |
| [**DataBuffer**](http://docs.google.com/java/awt/image/DataBuffer.html) | This class exists to wrap one or more data arrays. |
| [**DataBufferByte**](http://docs.google.com/java/awt/image/DataBufferByte.html) | This class extends DataBuffer and stores data internally as bytes. |
| [**DataBufferDouble**](http://docs.google.com/java/awt/image/DataBufferDouble.html) | This class extends DataBuffer and stores data internally in double form. |
| [**DataBufferFloat**](http://docs.google.com/java/awt/image/DataBufferFloat.html) | This class extends DataBuffer and stores data internally in float form. |
| [**DataBufferInt**](http://docs.google.com/java/awt/image/DataBufferInt.html) | This class extends DataBuffer and stores data internally as integers. |
| [**DataBufferShort**](http://docs.google.com/java/awt/image/DataBufferShort.html) | This class extends DataBuffer and stores data internally as shorts. |
| [**DataBufferUShort**](http://docs.google.com/java/awt/image/DataBufferUShort.html) | This class extends DataBuffer and stores data internally as shorts. |
| [**DirectColorModel**](http://docs.google.com/java/awt/image/DirectColorModel.html) | The DirectColorModel class is a ColorModel class that works with pixel values that represent RGB color and alpha information as separate samples and that pack all samples for a single pixel into a single int, short, or byte quantity. |
| [**FilteredImageSource**](http://docs.google.com/java/awt/image/FilteredImageSource.html) | This class is an implementation of the ImageProducer interface which takes an existing image and a filter object and uses them to produce image data for a new filtered version of the original image. |
| [**ImageFilter**](http://docs.google.com/java/awt/image/ImageFilter.html) | This class implements a filter for the set of interface methods that are used to deliver data from an ImageProducer to an ImageConsumer. |
| [**IndexColorModel**](http://docs.google.com/java/awt/image/IndexColorModel.html) | The IndexColorModel class is a ColorModel class that works with pixel values consisting of a single sample that is an index into a fixed colormap in the default sRGB color space. |
| [**Kernel**](http://docs.google.com/java/awt/image/Kernel.html) | The Kernel class defines a matrix that describes how a specified pixel and its surrounding pixels affect the value computed for the pixel's position in the output image of a filtering operation. |
| [**LookupOp**](http://docs.google.com/java/awt/image/LookupOp.html) | This class implements a lookup operation from the source to the destination. |
| [**LookupTable**](http://docs.google.com/java/awt/image/LookupTable.html) | This abstract class defines a lookup table object. |
| [**MemoryImageSource**](http://docs.google.com/java/awt/image/MemoryImageSource.html) | This class is an implementation of the ImageProducer interface which uses an array to produce pixel values for an Image. |
| [**MultiPixelPackedSampleModel**](http://docs.google.com/java/awt/image/MultiPixelPackedSampleModel.html) | The MultiPixelPackedSampleModel class represents one-banded images and can pack multiple one-sample pixels into one data element. |
| [**PackedColorModel**](http://docs.google.com/java/awt/image/PackedColorModel.html) | The PackedColorModel class is an abstract [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) class that works with pixel values which represent color and alpha information as separate samples and which pack all samples for a single pixel into a single int, short, or byte quantity. |
| [**PixelGrabber**](http://docs.google.com/java/awt/image/PixelGrabber.html) | The PixelGrabber class implements an ImageConsumer which can be attached to an Image or ImageProducer object to retrieve a subset of the pixels in that image. |
| [**PixelInterleavedSampleModel**](http://docs.google.com/java/awt/image/PixelInterleavedSampleModel.html) | This class represents image data which is stored in a pixel interleaved fashion and for which each sample of a pixel occupies one data element of the DataBuffer. |
| [**Raster**](http://docs.google.com/java/awt/image/Raster.html) | A class representing a rectangular array of pixels. |
| [**ReplicateScaleFilter**](http://docs.google.com/java/awt/image/ReplicateScaleFilter.html) | An ImageFilter class for scaling images using the simplest algorithm. |
| [**RescaleOp**](http://docs.google.com/java/awt/image/RescaleOp.html) | This class performs a pixel-by-pixel rescaling of the data in the source image by multiplying the sample values for each pixel by a scale factor and then adding an offset. |
| [**RGBImageFilter**](http://docs.google.com/java/awt/image/RGBImageFilter.html) | This class provides an easy way to create an ImageFilter which modifies the pixels of an image in the default RGB ColorModel. |
| [**SampleModel**](http://docs.google.com/java/awt/image/SampleModel.html) | This abstract class defines an interface for extracting samples of pixels in an image. |
| [**ShortLookupTable**](http://docs.google.com/java/awt/image/ShortLookupTable.html) | This class defines a lookup table object. |
| [**SinglePixelPackedSampleModel**](http://docs.google.com/java/awt/image/SinglePixelPackedSampleModel.html) | This class represents pixel data packed such that the N samples which make up a single pixel are stored in a single data array element, and each data data array element holds samples for only one pixel. |
| [**VolatileImage**](http://docs.google.com/java/awt/image/VolatileImage.html) | VolatileImage is an image which can lose its contents at any time due to circumstances beyond the control of the application (e.g., situations caused by the operating system or by other applications). |
| [**WritableRaster**](http://docs.google.com/java/awt/image/WritableRaster.html) | This class extends Raster to provide pixel writing capabilities. |

| **Exception Summary** | |
| --- | --- |
| [**ImagingOpException**](http://docs.google.com/java/awt/image/ImagingOpException.html) | The ImagingOpException is thrown if one of the [BufferedImageOp](http://docs.google.com/java/awt/image/BufferedImageOp.html) or [RasterOp](http://docs.google.com/java/awt/image/RasterOp.html) filter methods cannot process the image. |
| [**RasterFormatException**](http://docs.google.com/java/awt/image/RasterFormatException.html) | The RasterFormatException is thrown if there is invalid layout information in the [Raster](http://docs.google.com/java/awt/image/Raster.html). |

## Package java.awt.image Description

Provides classes for creating and modifying images. Images are processed using a streaming framework that involves an image producer, optional image filters, and an image consumer. This framework makes it possible to progressively render an image while it is being fetched and generated. Moreover, the framework allows an application to discard the storage used by an image and to regenerate it at any time. This package provides a number of image producers, consumers, and filters that you can configure for your image processing needs.

**Since:** JDK1.0

| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.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 PACKAGE**](http://docs.google.com/java/awt/im/spi/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/java/awt/image/renderable/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[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).