| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/PixelGrabber.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/java/awt/image/PackedColorModel.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/image/PixelInterleavedSampleModel.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/PixelGrabber.html)    [**NO FRAMES**](http://docs.google.com/PixelGrabber.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#4d34og8) | [METHOD](#26in1rg) |

## **java.awt.image**

Class PixelGrabber

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **java.awt.image.PixelGrabber**

**All Implemented Interfaces:** [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html)

public class **PixelGrabber**extends [Object](http://docs.google.com/java/lang/Object.html)implements [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.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. Here is an example:

public void handlesinglepixel(int x, int y, int pixel) {  
 int alpha = (pixel >> 24) & 0xff;  
 int red = (pixel >> 16) & 0xff;  
 int green = (pixel >> 8) & 0xff;  
 int blue = (pixel ) & 0xff;  
 // Deal with the pixel as necessary...  
 }  
  
 public void handlepixels(Image img, int x, int y, int w, int h) {  
 int[] pixels = new int[w \* h];  
 PixelGrabber pg = new PixelGrabber(img, x, y, w, h, pixels, 0, w);  
 try {  
 pg.grabPixels();  
 } catch (InterruptedException e) {  
 System.err.println("interrupted waiting for pixels!");  
 return;  
 }  
 if ((pg.getStatus() & ImageObserver.ABORT) != 0) {  
 System.err.println("image fetch aborted or errored");  
 return;  
 }  
 for (int j = 0; j < h; j++) {  
 for (int i = 0; i < w; i++) {  
 handlesinglepixel(x+i, y+j, pixels[j \* w + i]);  
 }  
 }  
 }

**See Also:**[ColorModel.getRGBdefault()](http://docs.google.com/java/awt/image/ColorModel.html#getRGBdefault())

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

| **Fields inherited from interface java.awt.image.**[**ImageConsumer**](http://docs.google.com/java/awt/image/ImageConsumer.html) |
| --- |
| [COMPLETESCANLINES](http://docs.google.com/java/awt/image/ImageConsumer.html#COMPLETESCANLINES), [IMAGEABORTED](http://docs.google.com/java/awt/image/ImageConsumer.html#IMAGEABORTED), [IMAGEERROR](http://docs.google.com/java/awt/image/ImageConsumer.html#IMAGEERROR), [RANDOMPIXELORDER](http://docs.google.com/java/awt/image/ImageConsumer.html#RANDOMPIXELORDER), [SINGLEFRAME](http://docs.google.com/java/awt/image/ImageConsumer.html#SINGLEFRAME), [SINGLEFRAMEDONE](http://docs.google.com/java/awt/image/ImageConsumer.html#SINGLEFRAMEDONE), [SINGLEPASS](http://docs.google.com/java/awt/image/ImageConsumer.html#SINGLEPASS), [STATICIMAGEDONE](http://docs.google.com/java/awt/image/ImageConsumer.html#STATICIMAGEDONE), [TOPDOWNLEFTRIGHT](http://docs.google.com/java/awt/image/ImageConsumer.html#TOPDOWNLEFTRIGHT) |

| **Constructor Summary** | |
| --- | --- |
| [**PixelGrabber**](http://docs.google.com/java/awt/image/PixelGrabber.html#PixelGrabber(java.awt.Image,%20int,%20int,%20int,%20int,%20boolean))([Image](http://docs.google.com/java/awt/Image.html) img, int x, int y, int w, int h, boolean forceRGB)            Create a PixelGrabber object to grab the (x, y, w, h) rectangular section of pixels from the specified image. |
| [**PixelGrabber**](http://docs.google.com/java/awt/image/PixelGrabber.html#PixelGrabber(java.awt.Image,%20int,%20int,%20int,%20int,%20int%5B%5D,%20int,%20int))([Image](http://docs.google.com/java/awt/Image.html) img, int x, int y, int w, int h, int[] pix, int off, int scansize)            Create a PixelGrabber object to grab the (x, y, w, h) rectangular section of pixels from the specified image into the given array. |
| [**PixelGrabber**](http://docs.google.com/java/awt/image/PixelGrabber.html#PixelGrabber(java.awt.image.ImageProducer,%20int,%20int,%20int,%20int,%20int%5B%5D,%20int,%20int))([ImageProducer](http://docs.google.com/java/awt/image/ImageProducer.html) ip, int x, int y, int w, int h, int[] pix, int off, int scansize)            Create a PixelGrabber object to grab the (x, y, w, h) rectangular section of pixels from the image produced by the specified ImageProducer into the given array. |

| **Method Summary** | |
| --- | --- |
| void | [**abortGrabbing**](http://docs.google.com/java/awt/image/PixelGrabber.html#abortGrabbing())()            Request the PixelGrabber to abort the image fetch. |
| [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) | [**getColorModel**](http://docs.google.com/java/awt/image/PixelGrabber.html#getColorModel())()            Get the ColorModel for the pixels stored in the array. |
| int | [**getHeight**](http://docs.google.com/java/awt/image/PixelGrabber.html#getHeight())()            Get the height of the pixel buffer (after adjusting for image height). |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getPixels**](http://docs.google.com/java/awt/image/PixelGrabber.html#getPixels())()            Get the pixel buffer. |
| int | [**getStatus**](http://docs.google.com/java/awt/image/PixelGrabber.html#getStatus())()            Return the status of the pixels. |
| int | [**getWidth**](http://docs.google.com/java/awt/image/PixelGrabber.html#getWidth())()            Get the width of the pixel buffer (after adjusting for image width). |
| boolean | [**grabPixels**](http://docs.google.com/java/awt/image/PixelGrabber.html#grabPixels())()            Request the Image or ImageProducer to start delivering pixels and wait for all of the pixels in the rectangle of interest to be delivered. |
| boolean | [**grabPixels**](http://docs.google.com/java/awt/image/PixelGrabber.html#grabPixels(long))(long ms)            Request the Image or ImageProducer to start delivering pixels and wait for all of the pixels in the rectangle of interest to be delivered or until the specified timeout has elapsed. |
| void | [**imageComplete**](http://docs.google.com/java/awt/image/PixelGrabber.html#imageComplete(int))(int status)            The imageComplete method is part of the ImageConsumer API which this class must implement to retrieve the pixels. |
| void | [**setColorModel**](http://docs.google.com/java/awt/image/PixelGrabber.html#setColorModel(java.awt.image.ColorModel))([ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model)            The setColorModel method is part of the ImageConsumer API which this class must implement to retrieve the pixels. |
| void | [**setDimensions**](http://docs.google.com/java/awt/image/PixelGrabber.html#setDimensions(int,%20int))(int width, int height)            The setDimensions method is part of the ImageConsumer API which this class must implement to retrieve the pixels. |
| void | [**setHints**](http://docs.google.com/java/awt/image/PixelGrabber.html#setHints(int))(int hints)            The setHints method is part of the ImageConsumer API which this class must implement to retrieve the pixels. |
| void | [**setPixels**](http://docs.google.com/java/awt/image/PixelGrabber.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int))(int srcX, int srcY, int srcW, int srcH, [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model, byte[] pixels, int srcOff, int srcScan)            The setPixels method is part of the ImageConsumer API which this class must implement to retrieve the pixels. |
| void | [**setPixels**](http://docs.google.com/java/awt/image/PixelGrabber.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20int%5B%5D,%20int,%20int))(int srcX, int srcY, int srcW, int srcH, [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model, int[] pixels, int srcOff, int srcScan)            The setPixels method is part of the ImageConsumer API which this class must implement to retrieve the pixels. |
| void | [**setProperties**](http://docs.google.com/java/awt/image/PixelGrabber.html#setProperties(java.util.Hashtable))([Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> props)            The setProperties method is part of the ImageConsumer API which this class must implement to retrieve the pixels. |
| void | [**startGrabbing**](http://docs.google.com/java/awt/image/PixelGrabber.html#startGrabbing())()            Request the PixelGrabber to start fetching the pixels. |
| int | [**status**](http://docs.google.com/java/awt/image/PixelGrabber.html#status())()            Returns the status of the pixels. |

| **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** |
| --- |

### PixelGrabber

public **PixelGrabber**([Image](http://docs.google.com/java/awt/Image.html) img,  
 int x,  
 int y,  
 int w,  
 int h,  
 int[] pix,  
 int off,  
 int scansize)

Create a PixelGrabber object to grab the (x, y, w, h) rectangular section of pixels from the specified image into the given array. The pixels are stored into the array in the default RGB ColorModel. The RGB data for pixel (i, j) where (i, j) is inside the rectangle (x, y, w, h) is stored in the array at pix[(j - y) \* scansize + (i - x) + off].

**Parameters:**img - the image to retrieve pixels fromx - the x coordinate of the upper left corner of the rectangle of pixels to retrieve from the image, relative to the default (unscaled) size of the imagey - the y coordinate of the upper left corner of the rectangle of pixels to retrieve from the imagew - the width of the rectangle of pixels to retrieveh - the height of the rectangle of pixels to retrievepix - the array of integers which are to be used to hold the RGB pixels retrieved from the imageoff - the offset into the array of where to store the first pixelscansize - the distance from one row of pixels to the next in the array**See Also:**[ColorModel.getRGBdefault()](http://docs.google.com/java/awt/image/ColorModel.html#getRGBdefault())

### PixelGrabber

public **PixelGrabber**([ImageProducer](http://docs.google.com/java/awt/image/ImageProducer.html) ip,  
 int x,  
 int y,  
 int w,  
 int h,  
 int[] pix,  
 int off,  
 int scansize)

Create a PixelGrabber object to grab the (x, y, w, h) rectangular section of pixels from the image produced by the specified ImageProducer into the given array. The pixels are stored into the array in the default RGB ColorModel. The RGB data for pixel (i, j) where (i, j) is inside the rectangle (x, y, w, h) is stored in the array at pix[(j - y) \* scansize + (i - x) + off].

**Parameters:**ip - the ImageProducer that produces the image from which to retrieve pixelsx - the x coordinate of the upper left corner of the rectangle of pixels to retrieve from the image, relative to the default (unscaled) size of the imagey - the y coordinate of the upper left corner of the rectangle of pixels to retrieve from the imagew - the width of the rectangle of pixels to retrieveh - the height of the rectangle of pixels to retrievepix - the array of integers which are to be used to hold the RGB pixels retrieved from the imageoff - the offset into the array of where to store the first pixelscansize - the distance from one row of pixels to the next in the array**See Also:**[ColorModel.getRGBdefault()](http://docs.google.com/java/awt/image/ColorModel.html#getRGBdefault())

### PixelGrabber

public **PixelGrabber**([Image](http://docs.google.com/java/awt/Image.html) img,  
 int x,  
 int y,  
 int w,  
 int h,  
 boolean forceRGB)

Create a PixelGrabber object to grab the (x, y, w, h) rectangular section of pixels from the specified image. The pixels are accumulated in the original ColorModel if the same ColorModel is used for every call to setPixels, otherwise the pixels are accumulated in the default RGB ColorModel. If the forceRGB parameter is true, then the pixels will be accumulated in the default RGB ColorModel anyway. A buffer is allocated by the PixelGrabber to hold the pixels in either case. If (w < 0) or (h < 0), then they will default to the remaining width and height of the source data when that information is delivered.

**Parameters:**img - the image to retrieve the image data fromx - the x coordinate of the upper left corner of the rectangle of pixels to retrieve from the image, relative to the default (unscaled) size of the imagey - the y coordinate of the upper left corner of the rectangle of pixels to retrieve from the imagew - the width of the rectangle of pixels to retrieveh - the height of the rectangle of pixels to retrieveforceRGB - true if the pixels should always be converted to the default RGB ColorModel

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

### startGrabbing

public void **startGrabbing**()

Request the PixelGrabber to start fetching the pixels.

### abortGrabbing

public void **abortGrabbing**()

Request the PixelGrabber to abort the image fetch.

### grabPixels

public boolean **grabPixels**()  
 throws [InterruptedException](http://docs.google.com/java/lang/InterruptedException.html)

Request the Image or ImageProducer to start delivering pixels and wait for all of the pixels in the rectangle of interest to be delivered.

**Returns:**true if the pixels were successfully grabbed, false on abort, error or timeout **Throws:** [InterruptedException](http://docs.google.com/java/lang/InterruptedException.html) - Another thread has interrupted this thread.

### grabPixels

public boolean **grabPixels**(long ms)  
 throws [InterruptedException](http://docs.google.com/java/lang/InterruptedException.html)

Request the Image or ImageProducer to start delivering pixels and wait for all of the pixels in the rectangle of interest to be delivered or until the specified timeout has elapsed. This method behaves in the following ways, depending on the value of ms:

* If ms == 0, waits until all pixels are delivered
* If ms > 0, waits until all pixels are delivered as timeout expires.
* If ms < 0, returns true if all pixels are grabbed, false otherwise and does not wait.

**Parameters:**ms - the number of milliseconds to wait for the image pixels to arrive before timing out **Returns:**true if the pixels were successfully grabbed, false on abort, error or timeout **Throws:** [InterruptedException](http://docs.google.com/java/lang/InterruptedException.html) - Another thread has interrupted this thread.

### getStatus

public int **getStatus**()

Return the status of the pixels. The ImageObserver flags representing the available pixel information are returned.

**Returns:**the bitwise OR of all relevant ImageObserver flags**See Also:**[ImageObserver](http://docs.google.com/java/awt/image/ImageObserver.html)

### getWidth

public int **getWidth**()

Get the width of the pixel buffer (after adjusting for image width). If no width was specified for the rectangle of pixels to grab then then this information will only be available after the image has delivered the dimensions.

**Returns:**the final width used for the pixel buffer or -1 if the width is not yet known**See Also:**[getStatus()](http://docs.google.com/java/awt/image/PixelGrabber.html#getStatus())

### getHeight

public int **getHeight**()

Get the height of the pixel buffer (after adjusting for image height). If no width was specified for the rectangle of pixels to grab then then this information will only be available after the image has delivered the dimensions.

**Returns:**the final height used for the pixel buffer or -1 if the height is not yet known**See Also:**[getStatus()](http://docs.google.com/java/awt/image/PixelGrabber.html#getStatus())

### getPixels

public [Object](http://docs.google.com/java/lang/Object.html) **getPixels**()

Get the pixel buffer. If the PixelGrabber was not constructed with an explicit pixel buffer to hold the pixels then this method will return null until the size and format of the image data is known. Since the PixelGrabber may fall back on accumulating the data in the default RGB ColorModel at any time if the source image uses more than one ColorModel to deliver the data, the array object returned by this method may change over time until the image grab is complete.

**Returns:**either a byte array or an int array**See Also:**[getStatus()](http://docs.google.com/java/awt/image/PixelGrabber.html#getStatus()), [setPixels(int, int, int, int, ColorModel, byte[], int, int)](http://docs.google.com/java/awt/image/PixelGrabber.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int)), [setPixels(int, int, int, int, ColorModel, int[], int, int)](http://docs.google.com/java/awt/image/PixelGrabber.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20int%5B%5D,%20int,%20int))

### getColorModel

public [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) **getColorModel**()

Get the ColorModel for the pixels stored in the array. If the PixelGrabber was constructed with an explicit pixel buffer then this method will always return the default RGB ColorModel, otherwise it may return null until the ColorModel used by the ImageProducer is known. Since the PixelGrabber may fall back on accumulating the data in the default RGB ColorModel at any time if the source image uses more than one ColorModel to deliver the data, the ColorModel object returned by this method may change over time until the image grab is complete and may not reflect any of the ColorModel objects that was used by the ImageProducer to deliver the pixels.

**Returns:**the ColorModel object used for storing the pixels**See Also:**[getStatus()](http://docs.google.com/java/awt/image/PixelGrabber.html#getStatus()), [ColorModel.getRGBdefault()](http://docs.google.com/java/awt/image/ColorModel.html#getRGBdefault()), [setColorModel(ColorModel)](http://docs.google.com/java/awt/image/PixelGrabber.html#setColorModel(java.awt.image.ColorModel))

### setDimensions

public void **setDimensions**(int width,  
 int height)

The setDimensions method is part of the ImageConsumer API which this class must implement to retrieve the pixels.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being grabbed. Developers using this class to retrieve pixels from an image should avoid calling this method directly since that operation could result in problems with retrieving the requested pixels.

**Specified by:**[setDimensions](http://docs.google.com/java/awt/image/ImageConsumer.html#setDimensions(int,%20int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**width - the width of the dimensionheight - the height of the dimension

### setHints

public void **setHints**(int hints)

The setHints method is part of the ImageConsumer API which this class must implement to retrieve the pixels.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being grabbed. Developers using this class to retrieve pixels from an image should avoid calling this method directly since that operation could result in problems with retrieving the requested pixels.

**Specified by:**[setHints](http://docs.google.com/java/awt/image/ImageConsumer.html#setHints(int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**hints - a set of hints used to process the pixels

### setProperties

public void **setProperties**([Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> props)

The setProperties method is part of the ImageConsumer API which this class must implement to retrieve the pixels.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being grabbed. Developers using this class to retrieve pixels from an image should avoid calling this method directly since that operation could result in problems with retrieving the requested pixels.

**Specified by:**[setProperties](http://docs.google.com/java/awt/image/ImageConsumer.html#setProperties(java.util.Hashtable)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**props - the list of properties

### setColorModel

public void **setColorModel**([ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model)

The setColorModel method is part of the ImageConsumer API which this class must implement to retrieve the pixels.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being grabbed. Developers using this class to retrieve pixels from an image should avoid calling this method directly since that operation could result in problems with retrieving the requested pixels.

**Specified by:**[setColorModel](http://docs.google.com/java/awt/image/ImageConsumer.html#setColorModel(java.awt.image.ColorModel)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**model - the specified ColorModel**See Also:**[getColorModel()](http://docs.google.com/java/awt/image/PixelGrabber.html#getColorModel())

### setPixels

public void **setPixels**(int srcX,  
 int srcY,  
 int srcW,  
 int srcH,  
 [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model,  
 byte[] pixels,  
 int srcOff,  
 int srcScan)

The setPixels method is part of the ImageConsumer API which this class must implement to retrieve the pixels.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being grabbed. Developers using this class to retrieve pixels from an image should avoid calling this method directly since that operation could result in problems with retrieving the requested pixels.

**Specified by:**[setPixels](http://docs.google.com/java/awt/image/ImageConsumer.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**srcX - the X coordinate of the upper-left corner of the area of pixels to be setsrcY - the Y coordinate of the upper-left corner of the area of pixels to be setsrcW - the width of the area of pixelssrcH - the height of the area of pixelsmodel - the specified ColorModelpixels - the array of pixelssrcOff - the offset into the pixels arraysrcScan - the distance from one row of pixels to the next in the pixels array**See Also:**[getPixels()](http://docs.google.com/java/awt/image/PixelGrabber.html#getPixels())

### setPixels

public void **setPixels**(int srcX,  
 int srcY,  
 int srcW,  
 int srcH,  
 [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model,  
 int[] pixels,  
 int srcOff,  
 int srcScan)

The setPixels method is part of the ImageConsumer API which this class must implement to retrieve the pixels.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being grabbed. Developers using this class to retrieve pixels from an image should avoid calling this method directly since that operation could result in problems with retrieving the requested pixels.

**Specified by:**[setPixels](http://docs.google.com/java/awt/image/ImageConsumer.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20int%5B%5D,%20int,%20int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**srcX - the X coordinate of the upper-left corner of the area of pixels to be setsrcY - the Y coordinate of the upper-left corner of the area of pixels to be setsrcW - the width of the area of pixelssrcH - the height of the area of pixelsmodel - the specified ColorModelpixels - the array of pixelssrcOff - the offset into the pixels arraysrcScan - the distance from one row of pixels to the next in the pixels array**See Also:**[getPixels()](http://docs.google.com/java/awt/image/PixelGrabber.html#getPixels())

### imageComplete

public void **imageComplete**(int status)

The imageComplete method is part of the ImageConsumer API which this class must implement to retrieve the pixels.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being grabbed. Developers using this class to retrieve pixels from an image should avoid calling this method directly since that operation could result in problems with retrieving the requested pixels.

**Specified by:**[imageComplete](http://docs.google.com/java/awt/image/ImageConsumer.html#imageComplete(int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**status - the status of image loading**See Also:**[ImageProducer.removeConsumer(java.awt.image.ImageConsumer)](http://docs.google.com/java/awt/image/ImageProducer.html#removeConsumer(java.awt.image.ImageConsumer))

### status

public int **status**()

Returns the status of the pixels. The ImageObserver flags representing the available pixel information are returned. This method and [getStatus](http://docs.google.com/java/awt/image/PixelGrabber.html#getStatus()) have the same implementation, but getStatus is the preferred method because it conforms to the convention of naming information-retrieval methods with the form "getXXX".

**Returns:**the bitwise OR of all relevant ImageObserver flags**See Also:**[ImageObserver](http://docs.google.com/java/awt/image/ImageObserver.html), [getStatus()](http://docs.google.com/java/awt/image/PixelGrabber.html#getStatus())

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/PixelGrabber.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/java/awt/image/PackedColorModel.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/image/PixelInterleavedSampleModel.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/PixelGrabber.html)    [**NO FRAMES**](http://docs.google.com/PixelGrabber.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#4d34og8) | [METHOD](#26in1rg) |

[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).