
Class java.awt.image.IndexColorModel

```
java.lang.Object
   |
   +----java.awt.image.ColorModel
           |
           +----java.awt.image.IndexColorModel
```

---

public class **IndexColorModel**
extends ColorModel

A ColorModel class that specifies a translation from pixel values to alpha, red, green, and blue color components for pixels which represent indices into a fixed colormap. An optional transparent pixel value can be supplied which indicates a completely transparent pixel, regardless of any alpha value recorded for that pixel value. This color model is similar to an X11 PseudoColor visual.

See Also:
ColorModel
Version:
1.8 10/07/95
Author:
Jim Graham

Constructor Index

o **IndexColorModel**(int, int, byte[], byte[], byte[])
Construct an IndexColorModel from the given arrays of red, green, and blue components.
o **IndexColorModel**(int, int, byte[], byte[], byte[], int)
Construct an IndexColorModel from the given arrays of red, green, and blue components.
o **IndexColorModel**(int, int, byte[], byte[], byte[], byte[])
Construct an IndexColorModel from the given arrays of red, green, blue and alpha components.
o **IndexColorModel**(int, int, byte[], int, boolean)
Construct an IndexColorModel from a single arrays of packed red, green, blue and optional alpha components.
o **IndexColorModel**(int, int, byte[], int, boolean, int)

Construct an IndexColorModel from a single arrays of packed red, green, blue and optional alpha components.

Method Index

o **getAlpha**(int)
Return the alpha transparency value for the specified pixel in the range 0–255.

o **getAlphas**(byte[])
Copy the array of alpha transparency values into the given array.

o **getBlue**(int)
Return the blue color compoment for the specified pixel in the range 0–255.

o **getBlues**(byte[])
Copy the array of blue color components into the given array.

o **getGreen**(int)
Return the green color compoment for the specified pixel in the range 0–255.

o **getGreens**(byte[])
Copy the array of green color components into the given array.

o **getMapSize**()
Returns the size of the color component arrays in this IndexColorModel.

o **getRGB**(int)
Return the color of the pixel in the default RGB color model.

o **getRed**(int)
Return the red color compoment for the specified pixel in the range 0–255.

o **getReds**(byte[])
Copy the array of red color components into the given array.

o **getTransparentPixel**()
Returns the index of the transparent pixel in this IndexColorModel or –1 if there is no transparent pixel.

Constructors

o IndexColorModel

```
public IndexColorModel(int bits,
                       int size,
                       byte r[],
                       byte g[],
                       byte b[])
```

Construct an IndexColorModel from the given arrays of red, green, and blue components. Pixels described by this color model will all have alpha components of 255 (fully opaque). All of the arrays specifying the color components must have at least the specified number of entries.

Parameters:

bits – The number of bits each pixel occupies.
size – The size of the color component arrays.
r – The array of red color components.
g – The array of green color components.

b – The array of blue color components.

o **IndexColorModel**

```
public IndexColorModel(int bits,
                       int size,
                       byte r[],
                       byte g[],
                       byte b[],
                       int trans)
```

Construct an IndexColorModel from the given arrays of red, green, and blue components. Pixels described by this color model will all have alpha components of 255 (fully opaque), except for the indicated transparent pixel. All of the arrays specifying the color components must have at least the specified number of entries.

Parameters:

bits – The number of bits each pixel occupies.
size – The size of the color component arrays.
r – The array of red color components.
g – The array of green color components.
b – The array of blue color components.
trans – The index of the transparent pixel.

o **IndexColorModel**

```
public IndexColorModel(int bits,
                       int size,
                       byte r[],
                       byte g[],
                       byte b[],
                       byte a[])
```

Construct an IndexColorModel from the given arrays of red, green, blue and alpha components. All of the arrays specifying the color components must have at least the specified number of entries.

Parameters:

bits – The number of bits each pixel occupies.
size – The size of the color component arrays.
r – The array of red color components.
g – The array of green color components.
b – The array of blue color components.
a – The array of alpha value components.

o **IndexColorModel**

```
public IndexColorModel(int bits,
                       int size,
                       byte cmap[],
                       int start,
                       boolean hasalpha)
```

Construct an IndexColorModel from a single arrays of packed red, green, blue and optional alpha components. The array must have enough values in it to fill all of the needed component arrays of the specified size.

Parameters:

bits – The number of bits each pixel occupies.
size – The size of the color component arrays.
cmap – The array of color components.
start – The starting offset of the first color component.
hasalpha – Indicates whether alpha values are contained in the cmap array.

o **IndexColorModel**

```
public IndexColorModel(int bits,
                       int size,
                       byte cmap[],
                       int start,
                       boolean hasalpha,
                       int trans)
```

Construct an IndexColorModel from a single arrays of packed red, green, blue and optional alpha components. The specified transparent index represents a pixel which will be considered entirely transparent regardless of any alpha value specified for it. The array must have enough values in it to fill all of the needed component arrays of the specified size.

Parameters:

bits – The number of bits each pixel occupies.
size – The size of the color component arrays.
cmap – The array of color components.
start – The starting offset of the first color component.
hasalpha – Indicates whether alpha values are contained in the cmap array.
trans – The index of the fully transparent pixel.

Methods

o **getMapSize**

```
public int getMapSize()
```

Returns the size of the color component arrays in this IndexColorModel.

o **getTransparentPixel**

```
public int getTransparentPixel()
```

Returns the index of the transparent pixel in this IndexColorModel or –1 if there is no transparent pixel.

o **getReds**

```
public void getReds(byte r[])
```

Copy the array of red color components into the given array. Only the initial entries of the array as specified by getMapSize() will be written.

o **getGreens**

```
public void getGreens(byte g[])
```

Copy the array of green color components into the given array. Only the initial entries of the array as specified by getMapSize() will be written.

o **getBlues**

```
public void getBlues(byte b[])
```

Copy the array of blue color components into the given array. Only the initial entries of the array as specified by getMapSize() will be written.

o **getAlphas**

```
public void getAlphas(byte a[])
```

Copy the array of alpha transparency values into the given array. Only the initial entries of the array as specified by getMapSize() will be written.

o **getRed**

```
public int getRed(int pixel)
```

Return the red color compoment for the specified pixel in the range 0–255.
Overrides:
getRed in class ColorModel

o **getGreen**

```
public int getGreen(int pixel)
```

Return the green color compoment for the specified pixel in the range 0–255.
Overrides:
getGreen in class ColorModel

o **getBlue**

```
public int getBlue(int pixel)
```

Return the blue color compoment for the specified pixel in the range 0–255.
Overrides:
getBlue in class ColorModel

o **getAlpha**

```
public int getAlpha(int pixel)
```

Return the alpha transparency value for the specified pixel in the range 0–255.
Overrides:
getAlpha in class ColorModel

o **getRGB**

```
public int getRGB(int pixel)
```

Return the color of the pixel in the default RGB color model.
Overrides:
getRGB in class ColorModel
See Also:
getRGBdefault
