NSColor Class Reference

Cocoa > Graphics & Imaging

2009-01-06

Contents

NSColor Class Reference

Inherits from	NSObject
Conforms to	NSCoding
	NSCopying
	NSObject (NSObject)
Framework	/System/Library/Frameworks/AppKit.framework
Availability	Available in Mac OS X v10.0 and later.
Companion guide	Color Programming Topics for Cocoa
Declared in	NSColor.h
Related sample code	NSGLImage
	OpenGL Screensaver
	Quartz Composer WWDC 2005 TextEdit
	Sketch-112
	TextEditPlus

Class at a Glance

An `NSColor` object represents a color, which is defined in a color space, each point of which has a set of components (such as red, green, and blue) that uniquely define a color.

Principal Attributes

- Color space
- Color components

  Various `colorWith...` and `colorUsing...` methods.
  Preset colors: `blackColor` (page 15), `blueColor` (page 15), and so on.

Commonly Used Methods

`colorUsingColorSpaceName:` (page 47)
　　　　Creates an `NSColor` instance in the specified color space.

`set` (page 57)
>   Sets the drawing color.

Overview

An `NSColor` object represents color and sometimes opacity (alpha). By sending a `set` (page 57) message to an `NSColor` instance, you set the color for the current drawing context. Setting the color causes subsequently drawn graphics to have the color represented by the `NSColor` instance.

It is invalid to use an accessor method related to components of a particular color space on an `NSColor` object that is not in that color space. For example, methods such as `redComponent` (page 56) and `getRed:green:blue:alpha:` (page 51) work on color objects in the calibrated and device RGB color spaces. If you send such a message to an `NSColor` object in the CMYK color space, an exception is raised. Further, the methods `getComponents:` (page 50) and `numberOfComponents` (page 56) work in color spaces that have individual components. Thus they return the components of `NSColor` objects as individual floating-point values regardless of whether they're based on `NSColorSpace` objects or named color spaces. However, older component-fetching methods such as `redComponent` and `getRed:green:blue:alpha:` are only effective on `NSColor` objects based on named color spaces.

If you have an `NSColor` object in an unknown color space and you want to extract its components, you should first convert the color object to a known color space before using the component accessor methods of that color space.

Adopted Protocols

NSCoding
>   `encodeWithCoder:`
>   `initWithCoder:`

NSCopying
>   `copyWithZone:`

Tasks

Creating an NSColor Object from Component Values

+ `colorWithCalibratedHue:saturation:brightness:alpha:` (page 17)
>   Creates and returns an `NSColor` object using the given opacity and HSB color space components.

+ `colorWithCalibratedRed:green:blue:alpha:` (page 18)
>   Creates and returns an `NSColor` object using the given opacity and RGB components.

+ `colorWithCalibratedWhite:alpha:` (page 19)
>   Creates and returns an `NSColor` object using the given opacity and grayscale value.

Creating an NSColor with Preset Components

+ whiteColor (page 40)

    Returns an NSColor object whose grayscale and alpha values are both 1.0.

+ yellowColor (page 41)

    Returns an NSColor object whose RGB value is 1.0, 1.0, 0.0 and whose alpha value is 1.0.

Working with Pattern Images

+ colorWithPatternImage: (page 24)

    Creates and returns an NSColor object that uses the specified image pattern.

– patternImage (page 56)

    Returns the image that the receiver is using as a pattern.

Creating a System Color—an NSColor Whose Value Is Specified by User Preferences

+ alternateSelectedControlColor (page 14)

    Returns the system color used for the face of a selected control.

+ alternateSelectedControlTextColor (page 14)

    Returns the system color used for text in a selected control.

+ colorForControlTint: (page 17)

    Returns the NSColor object specified by the given control tint.

+ controlBackgroundColor (page 25)

    Returns the system color used for the background of large controls.

+ controlColor (page 25)

    Returns the system color used for the flat surfaces of a control.

+ controlAlternatingRowBackgroundColors (page 24)

    Returns an array containing the system specified background colors for alternating rows in tables and lists.

+ controlHighlightColor (page 26)

    Returns the system color used for the highlighted bezels of controls.

+ controlLightHighlightColor (page 26)

    Returns the system color used for light highlights in controls.

+ controlShadowColor (page 27)

    Returns the system color used for the shadows dropped from controls.

+ controlDarkShadowColor (page 26)

    Returns the system color used for the dark edge of the shadow dropped from controls.

+ controlTextColor (page 27)

    Returns the system color used for text on controls that aren't disabled.

+ currentControlTint (page 28)

    Returns the current system control tint.

+ disabledControlTextColor (page 29)

    Returns the system color used for text on disabled controls.

Ignoring Alpha Components

+ `ignoresAlpha` (page 32)
    Returns a Boolean value indicating whether the application supports alpha.

+ `setIgnoresAlpha:` (page 38)
    Specifies whether an application supports alpha.

Copying and Pasting

+ `colorFromPasteboard:` (page 17)
    Returns the `NSColor` currently on the given pasteboard.

− `writeToPasteboard:` (page 59)
    Writes the receiver's data to the specified pasteboard.

Retrieving a Set of Components

− `getCyan:magenta:yellow:black:alpha:` (page 50)
    Returns the receiver's CMYK and opacity values.

− `getHue:saturation:brightness:alpha:` (page 51)
    Returns the receiver's HSB component and opacity values in the respective arguments.

− `getRed:green:blue:alpha:` (page 51)
    Returns the receiver's RGB component and opacity values in the respective arguments.

− `getWhite:alpha:` (page 52)
    Returns the receiver's grayscale value and alpha values.

− `getComponents:` (page 50)
    Returns the components of the receiver as an array.

− `numberOfComponents` (page 56)
    Returns the number of components in the receiver.

Retrieving Individual Components

− `alphaComponent` (page 42)
    Returns the receiver's alpha (opacity) component.

− `blackComponent` (page 42)
    Returns the receiver's black component.

− `blueComponent` (page 43)
    Returns the receiver's blue component.

− `brightnessComponent` (page 44)
    Returns the brightness component of the HSB color equivalent to the receiver.

− `catalogNameComponent` (page 44)
    Returns the name of the catalog containing the receiver's name.

− `colorNameComponent` (page 45)
    Returns the receiver's name.

Working with the Color Space

Changing the Color

- highlightWithLevel: (page 53)

  Returns an NSColor object that represents a blend between the receiver and the highlight color returned by highlightColor (page 31).
- shadowWithLevel: (page 58)

  Returns an NSColor object that represents a blend between the receiver and the shadow color returned by shadowColor (page 39).

Drawing

- drawSwatchInRect: (page 49)

  Draws the current color in the given rectangle.
- set (page 57)

  Sets the color of subsequent drawing to the color that the receiver represents.
- setFill (page 58)

  Sets the fill color of subsequent drawing to the receiver's color.
- setStroke (page 58)

  Sets the stroke color of subsequent drawing to the receiver's color.

Class Methods

alternateSelectedControlColor

Returns the system color used for the face of a selected control.

```
+ (NSColor *)alternateSelectedControlColor
```

Return Value
The system color used for the face of a selected control—a control being clicked or dragged. This color can be used where iApp-like highlighting is desired. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.2 and later.

See Also
+ alternateSelectedControlTextColor (page 14)
+ selectedControlColor (page 36)

Declared In
NSColor.h

alternateSelectedControlTextColor

Returns the system color used for text in a selected control.

```
+ (NSColor *)alternateSelectedControlTextColor
```

Return Value

The system color used for text in a selected control—a control being clicked or dragged. This color can be used where iApp-like highlighting is desired. For general information about system colors, see Accessing System Colors.

Availability

Available in Mac OS X v10.2 and later.

See Also

+ `alternateSelectedControlColor` (page 14)
+ `selectedControlTextColor` (page 36)

Declared In

`NSColor.h`

blackColor

Returns an `NSColor` object whose grayscale value is 0.0 and whose alpha value is 1.0.

`+ (NSColor *)blackColor`

Return Value

The `NSColor` object.

Availability

Available in Mac OS X v10.0 and later.

See Also

– `blackComponent` (page 42)

Related Sample Code

Dicey
DockTile
PDF Annotation Editor
SpeedometerView
WebKitPluginStarter

Declared In

`NSColor.h`

blueColor

Returns an `NSColor` object whose RGB value is 0.0, 0.0, 1.0 and whose alpha value is 1.0.

`+ (NSColor *)blueColor`

Return Value

The `NSColor` object.

Availability

Available in Mac OS X v10.0 and later.

See Also
- blueComponent (page 43)

Related Sample Code
CocoaVideoFrameToGWorld

FunkyOverlayWindow

Grady

SpeedometerView

VertexPerformanceTest

Declared In
NSColor.h

brownColor

Returns an NSColor object whose RGB value is 0.6, 0.4, 0.2 and whose alpha value is 1.0.

+ (NSColor *)brownColor

Return Value
The NSColor object.

Availability
Available in Mac OS X v10.0 and later.

Declared In
NSColor.h

clearColor

Returns an NSColor object whose grayscale and alpha values are both 0.0.

+ (NSColor *)clearColor

Return Value
The NSColor object.

Availability
Available in Mac OS X v10.0 and later.

Related Sample Code
FunkyOverlayWindow

GLChildWindowDemo

RoundTransparentWindow

Sketch-112

TrackBall

Declared In
NSColor.h

colorForControlTint:

Returns the `NSColor` object specified by the given control tint.

`+ (NSColor *)colorForControlTint:(NSControlTint)`*controlTint*

Parameters

controlTint

      The control tint for which to return an `NSColor` object. This is one of the tint settings. For more on control tints, see Using the System Control Tint.

Return Value

The `NSColor` object.

Availability

Available in Mac OS X v10.0 and later.

See Also

+ currentControlTint (page 28)

Declared In

NSColor.h

colorFromPasteboard:

Returns the `NSColor` currently on the given pasteboard.

`+ (NSColor *)colorFromPasteboard:(NSPasteboard *)`*pasteBoard*

Parameters

pasteBoard

      The pasteboard from which to return the color.

Return Value

The color currently on the pasteboard or `nil` if *pasteBoard* doesn't contain color data. The returned color's alpha component is set to 1.0 if `ignoresAlpha` (page 32) returns `YES`.

Availability

Available in Mac OS X v10.0 and later.

See Also

– writeToPasteboard: (page 59)

Related Sample Code

Sketch-112

Declared In

NSColor.h

colorWithCalibratedHue:saturation:brightness:alpha:

Creates and returns an `NSColor` object using the given opacity and HSB color space components.

```
+ (NSColor *)colorWithCalibratedHue:(CGFloat)hue saturation:(CGFloat)saturation
    brightness:(CGFloat)brightness alpha:(CGFloat)alpha
```

Parameters

hue

> The hue component of the color object in the HSB color space.

saturation

> The saturation component of the color object in the HSB color space.

brightness

> The brightness (or value) component of the color object in the HSB color space.

alpha

> The opacity value of the color object,

Return Value

The color object.

Discussion

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0.

Availability

Available in Mac OS X v10.0 and later.

See Also

+ colorWithCalibratedRed:green:blue:alpha: (page 18)

+ colorWithDeviceHue:saturation:brightness:alpha: (page 22)

– getHue:saturation:brightness:alpha: (page 51)

Related Sample Code

NewsReader

Declared In

NSColor.h

colorWithCalibratedRed:green:blue:alpha:

Creates and returns an `NSColor` object using the given opacity and RGB components.

```
+ (NSColor *)colorWithCalibratedRed:(CGFloat)red green:(CGFloat)green
    blue:(CGFloat)blue alpha:(CGFloat)alpha
```

Parameters

red

> The red component of the color object.

green

> The green component of the color object.

blue

> The blue component of the color object.

alpha

> The opacity value of the color object.

Return Value

The color object.

Discussion
Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ `colorWithCalibratedHue:saturation:brightness:alpha:` (page 17)
+ `colorWithDeviceRed:green:blue:alpha:` (page 23)
– `getRed:green:blue:alpha:` (page 51)

Related Sample Code
Color Sampler
DockTile
GLChildWindowDemo
Squiggles
WhackedTV

Declared In
`NSColor.h`

colorWithCalibratedWhite:alpha:

Creates and returns an `NSColor` object using the given opacity and grayscale value.

+ (NSColor *)`colorWithCalibratedWhite:`(CGFloat)*white* `alpha:`(CGFloat)*alpha*

Parameters
white
     The grayscale value of the color object.

alpha
     The opacity value of the color object.

Return Value
The color object.

Discussion
Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ `colorWithDeviceWhite:alpha:` (page 23)
– `getWhite:alpha:` (page 52)

Related Sample Code
Aperture Edit Plugin - Borders & Titles
Clock Control
iSpend
TrackBall
WhackedTV

Declared In
`NSColor.h`

colorWithCatalogName:colorName:

Creates and returns an `NSColor` object by finding the color with the specified name in the given catalog.

```
+ (NSColor *)colorWithCatalogName:(NSString *)listName colorName:(NSString
    *)colorName
```

Parameters

listName
> The name of the catalog in which to find the specified color; this may be a standard catalog.

colorName
> The name of the color. Note that the color must be defined in the named color space to retrieve it with this method.

Return Value
The color object.

Availability
Available in Mac OS X v10.0 and later.

See Also
- `catalogNameComponent` (page 44)
- `colorNameComponent` (page 45)
- `localizedCatalogNameComponent` (page 54)

Declared In
`NSColor.h`

colorWithCIColor:

Converts a Core Image color object to its `NSColor` equivalent.

```
+ (NSColor *)colorWithCIColor:(CIColor *)color
```

Parameters

color
> The Core Image color to convert.

Return Value
The `NSColor` object corresponding to the specified Core Image color.

Discussion
The method raises if the color space and components associated with *color* are `nil` or invalid.

Availability
Available in Mac OS X v10.4 and later.

Declared In
`NSColor.h`

colorWithColorSpace:components:count:

Returns an `NSColor` object created from the specified components of the given color space.

```
+ (NSColor *)colorWithColorSpace:(NSColorSpace *)space components:(const CGFloat
    *)components count:(NSInteger)numberOfComponents
```

Parameters

space

    An `NSColorSpace` object representing a color space. The method raises if this is `nil`.

components

    An array of the components in the specified color space to use to create the `NSColor` object. The order of these components is determined by the color-space profile, with the alpha component always last. (If you want the created color to be opaque, specify 1.0 for the alpha component.)

numberOfComponents

    The number of components in the `components` array. This should match the number dictated by the specified color space plus one for alpha. This method raises an exception if they do not match.

Return Value

The color object. If *space* represents a color space that cannot cannot be used with `NSColor` objects—for example, a "pattern" color space—the method returns `nil`.

Availability

Available in Mac OS X v10.4 and later.

See Also

– `colorUsingColorSpace:` (page 46)

Declared In

`NSColor.h`

colorWithDeviceCyan:magenta:yellow:black:alpha:

Creates and returns an `NSColor` object using the given opacity value and CMYK components.

```
+ (NSColor *)colorWithDeviceCyan:(CGFloat)cyan magenta:(CGFloat)magenta
    yellow:(CGFloat)yellow black:(CGFloat)black alpha:(CGFloat)alpha
```

Parameters

cyan

    The cyan component of the color object.

magenta

    The magenta component of the color object.

yellow

    The yellow component of the color object.

black

    The black component of the color object.

alpha

    The opacity value of the color object.

Return Value

The color object.

Discussion
Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0. In PostScript, this color space corresponds directly to the device-dependent operator `setcmykcolor`.

Availability
Available in Mac OS X v10.0 and later.

See Also
– `getCyan:magenta:yellow:black:alpha:` (page 50)

Declared In
`NSColor.h`

colorWithDeviceHue:saturation:brightness:alpha:

Creates and returns an `NSColor` object using the given opacity value and HSB color space components.

```
+ (NSColor *)colorWithDeviceHue:(CGFloat)hue saturation:(CGFloat)saturation
    brightness:(CGFloat)brightness alpha:(CGFloat)alpha
```

Parameters

hue
    The hue component of the color object.

saturation
    The saturation component of the color object.

brightness
    The brightness component of the color object.

alpha
    The opacity value of the color object.

Return Value
The color object.

Discussion
Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0. In PostScript, this color space corresponds directly to the device-dependent operator `setrgbcolor`.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ `colorWithCalibratedHue:saturation:brightness:alpha:` (page 17)
+ `colorWithDeviceRed:green:blue:alpha:` (page 23)
– `getHue:saturation:brightness:alpha:` (page 51)

Related Sample Code
MenuItemView

Declared In
`NSColor.h`

colorWithDeviceRed:green:blue:alpha:

Creates and returns an `NSColor` object using the given opacity value and RGB components.

```
+ (NSColor *)colorWithDeviceRed:(CGFloat)red green:(CGFloat)green blue:(CGFloat)blue
    alpha:(CGFloat)alpha
```

Parameters

red

      The red component of the color object.

green

      The green component of the color object.

blue

      The blue component of the color object.

alpha

      The opacity value of the color object.

Return Value

The color object.

Discussion

Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0. In PostScript, this color space corresponds directly to the device-dependent operator `setrgbcolor`.

Availability

Available in Mac OS X v10.0 and later.

See Also

+ `colorWithCalibratedRed:green:blue:alpha:` (page 18)

+ `colorWithDeviceHue:saturation:brightness:alpha:` (page 22)

– `getRed:green:blue:alpha:` (page 51)

Related Sample Code

NSGLImage

OpenGL Screensaver

OpenGLCompositorLab

PDF Annotation Editor

SonOfSillyBalls

Declared In

NSColor.h

colorWithDeviceWhite:alpha:

Creates and returns an `NSColor` object using the given opacity and grayscale values.

```
+ (NSColor *)colorWithDeviceWhite:(CGFloat)white alpha:(CGFloat)alpha
```

Parameters

white

      The grayscale value of the color object.

alpha
> The opacity value of the color object.

Return Value
The color object.

Discussion
Values below 0.0 are interpreted as 0.0, and values above 1.0 are interpreted as 1.0. In PostScript, this color space corresponds directly to the device-dependent operator `setgray`.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ `colorWithCalibratedWhite:alpha:` (page 19)
– `getWhite:alpha:` (page 52)

Related Sample Code
FilterDemo
Link Snoop
PDFKitLinker2

Declared In
NSColor.h

colorWithPatternImage:

Creates and returns an `NSColor` object that uses the specified image pattern.

+ (NSColor *)colorWithPatternImage:(NSImage *)*image*

Parameters
image
> The image to use as the pattern for the color object. The image is tiled starting at the bottom of the window. The image is not scaled.

Return Value
The `NSColor` object. This color object is autoreleased.

Availability
Available in Mac OS X v10.0 and later.

Related Sample Code
Dicey
FilterDemo
Worm

Declared In
NSColor.h

controlAlternatingRowBackgroundColors

Returns an array containing the system specified background colors for alternating rows in tables and lists.

```
+ (NSArray *)controlAlternatingRowBackgroundColors
```

Return Value
An array of `NSColor` objects specifying the system colors used for rows in tables and lists. You should not assume the array will contain only two colors. For general information on system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.3 and later.

Declared In
`NSColor.h`

controlBackgroundColor

Returns the system color used for the background of large controls.

```
+ (NSColor *)controlBackgroundColor
```

Return Value
The system color used for the background of large controls such as browsers, table views, and clip views. For general information on system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

Declared In
`NSColor.h`

controlColor

Returns the system color used for the flat surfaces of a control.

```
+ (NSColor *)controlColor
```

Return Value
The system color used for the flat surfaces of a control. By default, the control color is a pattern color that will draw the ruled lines for the window background, which is the same as returned by `windowBackgroundColor` (page 40).

If you use `controlColor` assuming that it is a solid, you may have an incorrect appearance. You should use `lightGrayColor` (page 33) in its place.

Availability
Available in Mac OS X v10.0 and later.

Related Sample Code
Quartz Composer WWDC 2005 TextEdit
TextEditPlus

Declared In
`NSColor.h`

controlDarkShadowColor

Returns the system color used for the dark edge of the shadow dropped from controls.

`+ (NSColor *)controlDarkShadowColor`

Return Value
Of the two dark borders that run along the bottom and right of controls, representing shadows, the color of the outer, darker border. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ controlShadowColor (page 27)

Related Sample Code
Sketch-112

Declared In
NSColor.h

controlHighlightColor

Returns the system color used for the highlighted bezels of controls.

`+ (NSColor *)controlHighlightColor`

Return Value
Of the two light borders that run along the top and left of controls, representing reflections from a light source in the upper left, the color of the inner, duller border. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ controlLightHighlightColor (page 26)

Declared In
NSColor.h

controlLightHighlightColor

Returns the system color used for light highlights in controls.

`+ (NSColor *)controlLightHighlightColor`

Return Value
Of the two light borders that run along the top and left of controls, representing reflections from a light source in the upper left, the color of the outer, brighter border. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ controlHighlightColor (page 26)

Declared In
NSColor.h

controlShadowColor

Returns the system color used for the shadows dropped from controls.

`+ (NSColor *)controlShadowColor`

Return Value
Of the two dark borders that run along the bottom and right of controls, representing shadows, the color of the inner, lighter border. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ controlDarkShadowColor (page 26)

Declared In
NSColor.h

controlTextColor

Returns the system color used for text on controls that aren't disabled.

`+ (NSColor *)controlTextColor`

Return Value
The color used for text on enabled controls. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ disabledControlTextColor (page 29)

Related Sample Code
NewsReader

Declared In
NSColor.h

currentControlTint

Returns the current system control tint.

```
+ (NSControlTint)currentControlTint
```

Return Value
The current system control tint.

Discussion
An application can register for the `NSControlTintDidChangeNotification` notification to be notified of changes to the system control tint.

Availability
Available in Mac OS X v10.3 and later.

See Also
+ colorForControlTint: (page 17)

Declared In
NSColor.h

cyanColor

Returns an `NSColor` object whose RGB value is 0.0, 1.0, 1.0 and whose alpha value is 1.0.

```
+ (NSColor *)cyanColor
```

Return Value
The `NSColor` object.

Availability
Available in Mac OS X v10.0 and later.

See Also
– cyanComponent (page 49)

Related Sample Code
LayoutManagerDemo

RGB Image

Declared In
NSColor.h

darkGrayColor

Returns an `NSColor` object whose grayscale value is 1/3 and whose alpha value is 1.0.

```
+ (NSColor *)darkGrayColor
```

Return Value
The `NSColor` object.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ lightGrayColor (page 33)
+ grayColor (page 29)

Related Sample Code
TextLinks
TrackIt

Declared In
NSColor.h

disabledControlTextColor

Returns the system color used for text on disabled controls.

+ (NSColor *)disabledControlTextColor

Return Value
The color used for text on disabled controls. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ controlTextColor (page 27)

Related Sample Code
NewsReader

Declared In
NSColor.h

grayColor

Returns an NSColor object whose grayscale value is 0.5 and whose alpha value is 1.0.

+ (NSColor *)grayColor

Return Value
The NSColor object.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ lightGrayColor (page 33)
+ darkGrayColor (page 28)

Related Sample Code
CocoaDragAndDrop
FilterDemo
ImageMapExample
PDFKitLinker2
URL CacheInfo

Declared In
NSColor.h

greenColor

Returns an NSColor object whose RGB value is 0.0, 1.0, 0.0 and whose alpha value is 1.0.

+ (NSColor *)greenColor

Return Value
The NSColor object.

Availability
Available in Mac OS X v10.0 and later.

See Also
– greenComponent (page 53)

Related Sample Code
DockTile
QTKitMovieShuffler
SpeedometerView
WebKitPluginStarter
WhackedTV

Declared In
NSColor.h

gridColor

Returns the system color used for the optional gridlines in, for example, a table view.

+ (NSColor *)gridColor

Return Value
The system color used for gridlines. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

Declared In
NSColor.h

headerColor

Returns the system color used as the background color for header cells in table views and outline views.

```
+ (NSColor *)headerColor
```

Return Value
The system color used as the background for header cells in table and outline views. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

Declared In
NSColor.h

headerTextColor

Returns the system color used for text in header cells in table views and outline views.

```
+ (NSColor *)headerTextColor
```

Return Value
The system color used for text in header cells in table and outline views. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

Declared In
NSColor.h

highlightColor

Returns the system color that represents the virtual light source on the screen.

```
+ (NSColor *)highlightColor
```

Return Value
The system color for the virtual light source on the screen.

Discussion
This method is invoked by the highlightWithLevel: (page 53) method. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
- highlightWithLevel: (page 53)

Declared In
NSColor.h

ignoresAlpha

Returns a Boolean value indicating whether the application supports alpha.

`+ (BOOL)ignoresAlpha`

Return Value
`YES` if the application doesn't support alpha; otherwise `NO`. This value is consulted when an application imports alpha (through color dragging, for instance). The value determines whether the color panel has an opacity slider.

This value is `YES` by default, indicating that the opacity components of imported colors will be set to 1.0. If an application wants alpha, it can invoke the `setIgnoresAlpha:` (page 38) method with a parameter of `NO`.

Availability
Available in Mac OS X v10.0 and later.

See Also
`+ setIgnoresAlpha:` (page 38)
`– alphaComponent` (page 42)

Declared In
`NSColor.h`

keyboardFocusIndicatorColor

Returns the system color that represents the keyboard focus ring around controls.

`+ (NSColor *)keyboardFocusIndicatorColor`

Return Value
The system color representing the focus ring.

Availability
Available in Mac OS X v10.0 and later.

Declared In
`NSColor.h`

knobColor

Returns the system color used for the flat surface of a slider knob that hasn't been selected.

`+ (NSColor *)knobColor`

Return Value
The system color used for an unselected slider knob.

Discussion
The knob's beveled edges, which set it in relief, are drawn in highlighted and shadowed versions of the face color. When a knob is selected, its color changes to `selectedKnobColor` (page 36). For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

Related Sample Code
Sketch-112

Declared In
`NSColor.h`

lightGrayColor

Returns an `NSColor` object whose grayscale value is 2/3 and whose alpha value is 1.0.

`+ (NSColor *)lightGrayColor`

Return Value
The `NSColor` object.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ `grayColor` (page 29)
+ `darkGrayColor` (page 28)

Related Sample Code
IdentitySample
Quartz Composer WWDC 2005 TextEdit
Sketch-112
TextEditPlus
TrackIt

Declared In
`NSColor.h`

magentaColor

Returns an `NSColor` object whose RGB value is 1.0, 0.0, 1.0 and whose alpha value is 1.0.

`+ (NSColor *)magentaColor`

Return Value
The `NSColor` object.

Availability
Available in Mac OS X v10.0 and later.

See Also
– `magentaComponent` (page 55)

Related Sample Code
CocoaVideoFrameToGWorld

LayoutManagerDemo
RGB Image

Declared In
NSColor.h

orangeColor

Returns an NSColor object whose RGB value is 1.0, 0.5, 0.0 and whose alpha value is 1.0.

```
+ (NSColor *)orangeColor
```

Return Value
The NSColor object.

Availability
Available in Mac OS X v10.0 and later.

Related Sample Code
Grady
MenuItemView

Declared In
NSColor.h

purpleColor

Returns an NSColor object whose RGB value is 0.5, 0.0, 0.5 and whose alpha value is 1.0.

```
+ (NSColor *)purpleColor
```

Return Value
The NSColor object.

Availability
Available in Mac OS X v10.0 and later.

Related Sample Code
MenuItemView

Declared In
NSColor.h

redColor

Returns an NSColor object whose RGB value is 1.0, 0.0, 0.0 and whose alpha value is 1.0.

```
+ (NSColor *)redColor
```

Return Value
The NSColor object.

Availability
Available in Mac OS X v10.0 and later.

See Also
– redComponent (page 56)

Related Sample Code
Dicey
DockTile
SpeedometerView
UIElementInspector
WhackedTV

Declared In
NSColor.h

scrollBarColor

Returns the system color used for scroll "bars"—that is, for the groove in which a scroller's knob moves

+ (NSColor *)scrollBarColor

Return Value
The system color used for scroll bars. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

Declared In
NSColor.h

secondarySelectedControlColor

Returns the system color used in non-key views.

+ (NSColor *)secondarySelectedControlColor

Return Value
The system color used in non-key views. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.1 and later.

See Also
+ selectedControlColor (page 36)

Declared In
NSColor.h

selectedControlColor

Returns the system color used for the face of a selected control.

`+ (NSColor *)selectedControlColor`

Return Value
The system color used for the face of a selected control—a control being dragged or clicked. For general information about system colors, see Accessing System Colors

Availability
Available in Mac OS X v10.0 and later.

See Also
+ selectedControlTextColor (page 36)
+ secondarySelectedControlColor (page 35)
+ alternateSelectedControlColor (page 14)

Declared In
NSColor.h

selectedControlTextColor

Returns the system color used for text in a selected control—a control being clicked or dragged.

`+ (NSColor *)selectedControlTextColor`

Return Value
The system color used for text in a selected control—a control being clicked or dragged. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ selectedControlColor (page 36)
+ alternateSelectedControlTextColor (page 14)

Declared In
NSColor.h

selectedKnobColor

Returns the system color used for the slider knob when it is selected.

`+ (NSColor *)selectedKnobColor`

Return Value
The system color used for a slider knob that is selected—that is, dragged. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

Declared In
NSColor.h

selectedMenuItemColor

Returns the system color used for the face of selected menu items.

+ (NSColor *)selectedMenuItemColor

Return Value
The system color used for selected menu items. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

Declared In
NSColor.h

selectedMenuItemTextColor

Returns the system color used for the text in menu items.

+ (NSColor *)selectedMenuItemTextColor

Return Value
The system color used for text in selected menu items. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

Declared In
NSColor.h

selectedTextBackgroundColor

Returns the system color used for the background of selected text.

+ (NSColor *)selectedTextBackgroundColor

Return Value
The system color used for the background of selected text. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ selectedTextColor (page 38)

Declared In
NSColor.h

selectedTextColor

Returns the system color used for selected text.

+ (NSColor *)selectedTextColor

Return Value
The system color used for selected text. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ selectedTextBackgroundColor (page 37)

Declared In
NSColor.h

setIgnoresAlpha:

Specifies whether an application supports alpha.

+ (void)setIgnoresAlpha:(BOOL)*flag*

Parameters
flag
> YES to indicate that the application won't support alpha. By default, applications ignore alpha.

Discussion
If the application doesn't support alpha, no opacity slider is displayed in the color panel, and colors dragged in or pasted have their alpha values set to 1.0. Applications that need to import alpha can invoke this method with *flag* set to NO and explicitly make colors opaque in cases where it matters to them. Note that calling this with a value of YES overrides any value set with the NSColorPanel method setShowsAlpha:.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ ignoresAlpha (page 32)

– `alphaComponent` (page 42)

Related Sample Code
Aperture Edit Plugin - Borders & Titles
Quartz Composer QCTV
Tinted Image

Declared In
`NSColor.h`

shadowColor

Returns the system color that represents the virtual shadows cast by raised objects on the screen.

`+ (NSColor *)shadowColor`

Return Value
The system color for the virtual shadows case by raised objects on the screen.

Discussion
This method is invoked by `shadowWithLevel:` (page 58). For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
– `shadowWithLevel:` (page 58)

Declared In
`NSColor.h`

textBackgroundColor

Returns the system color used for the text background.

`+ (NSColor *)textBackgroundColor`

Return Value
The system color used for the background of text. When text is selected, its background color changes to the return value of `selectedTextBackgroundColor` (page 37). For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
`+ textColor` (page 40)

Declared In
`NSColor.h`

textColor

Returns the system color used for text.

```
+ (NSColor *)textColor
```

Return Value
The system color used for text. When text is selected, its color changes to the return value of
selectedTextColor (page 38). For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ textBackgroundColor (page 39)

Declared In
NSColor.h

whiteColor

Returns an NSColor object whose grayscale and alpha values are both 1.0.

```
+ (NSColor *)whiteColor
```

Return Value
The NSColor object.

Availability
Available in Mac OS X v10.0 and later.

See Also
– whiteComponent (page 59)

Related Sample Code
Dicey
OpenGL Screensaver
Quartz Composer WWDC 2005 TextEdit
Sketch-112
TextEditPlus

Declared In
NSColor.h

windowBackgroundColor

Returns a pattern color that will draw the ruled lines for the window background.

```
+ (NSColor *)windowBackgroundColor
```

Return Value
The pattern color used for the background of a window. For general information about system colors, see
Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

Related Sample Code
JDragNDrop

Declared In
`NSColor.h`

windowFrameColor

Returns the system color used for window frames, except for their text.

`+ (NSColor *)windowFrameColor`

Return Value
The system color used for window frames. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
`+ windowFrameTextColor` (page 41)

Declared In
`NSColor.h`

windowFrameTextColor

Returns the system color used for the text in window frames.

`+ (NSColor *)windowFrameTextColor`

Return Value
The system color used for text in window frames. For general information about system colors, see Accessing System Colors.

Availability
Available in Mac OS X v10.0 and later.

See Also
`+ windowFrameColor` (page 41)

Declared In
`NSColor.h`

yellowColor

Returns an `NSColor` object whose RGB value is 1.0, 1.0, 0.0 and whose alpha value is 1.0.

`+ (NSColor *)yellowColor`

Return Value
The `NSColor` object.

Availability
Available in Mac OS X v10.0 and later.

See Also
- `yellowComponent` (page 60)

Related Sample Code
Dicey
DockTile
LayoutManagerDemo
SpeedometerView
StickiesExample

Declared In
`NSColor.h`

Instance Methods

alphaComponent

Returns the receiver's alpha (opacity) component.

    - (CGFloat)alphaComponent

Return Value
The alpha component of the color object. If the receiver has no alpha component, this is 1.0 (opaque).

Availability
Available in Mac OS X v10.0 and later.

See Also
- `getCyan:magenta:yellow:black:alpha:` (page 50)
- `getHue:saturation:brightness:alpha:` (page 51)
- `getRed:green:blue:alpha:` (page 51)
- `getWhite:alpha:` (page 52)

Related Sample Code
Monochrome Image

Declared In
`NSColor.h`

blackComponent

Returns the receiver's black component.

    - (CGFloat)blackComponent

Return Value
The color object's black component.

Discussion
This method works only with objects representing colors in the `NSDeviceCMYKColorSpace` color space. Sending it to other objects raises an exception.

Availability
Available in Mac OS X v10.0 and later.

See Also
– `getCyan:magenta:yellow:black:alpha:` (page 50)

Declared In
`NSColor.h`

blendedColorWithFraction:ofColor:

Creates and returns an `NSColor` object whose component values are a weighted sum of the receiver's and the specified color object's.

```
- (NSColor *)blendedColorWithFraction:(CGFloat)fraction ofColor:(NSColor *)color
```

Parameters

fraction

    The amount of the color to blend with the receiver's color. The method converts *color* and a copy of the receiver to RGB, and then sets each component of the returned color to *fraction* of *color*'s value plus 1 – *fraction* of the receiver's.

color

    The color to blend with the receiver's color.

Return Value
The resulting color object or `nil` if the colors can't be converted.

Availability
Available in Mac OS X v10.0 and later.

Related Sample Code
DockTile
SpeedometerView
WebKitPluginStarter
WebKitPluginWithJavaScript

Declared In
`NSColor.h`

blueComponent

Returns the receiver's blue component.

```
- (CGFloat)blueComponent
```

Return Value
The color object's blue component.

Discussion
This method works only with objects representing colors in the `NSCalibratedRGBColorSpace` or `NSDeviceRGBColorSpace` color space. Sending it to other objects raises an exception.

Availability
Available in Mac OS X v10.0 and later.

See Also
– `getRed:green:blue:alpha:` (page 51)

Related Sample Code
Color Sampler
QTCoreVideo301

Declared In
`NSColor.h`

brightnessComponent

Returns the brightness component of the HSB color equivalent to the receiver.

– `(CGFloat)brightnessComponent`

Return Value
The color object's brightness component.

Discussion
This method works only with objects representing colors in the `NSCalibratedRGBColorSpace` or `NSDeviceRGBColorSpace` color space. Sending it to other objects raises an exception.

Availability
Available in Mac OS X v10.0 and later.

See Also
– `getHue:saturation:brightness:alpha:` (page 51)

Declared In
`NSColor.h`

catalogNameComponent

Returns the name of the catalog containing the receiver's name.

– `(NSString *)catalogNameComponent`

Return Value
The name of the catalog containing the color object.

Discussion
This method raises an exception if the receiver's color space isn't `NSNamedColorSpace`.

Availability
Available in Mac OS X v10.0 and later.

See Also

Declared In
NSColor.h

colorNameComponent

Returns the receiver's name.

    - (NSString *)colorNameComponent

Return Value
The name of the color object.

Discussion
This method raises an exception if the receiver's color space isn't NSNamedColorSpace.

Availability
Available in Mac OS X v10.0 and later.

See Also

Declared In
NSColor.h

colorSpace

Returns an object representing the color space of the receiver.

    - (NSColorSpace *)colorSpace

Return Value
An object representing a color space. The returned NSColorSpace object may represent a custom color space.

Discussion
Calling this method raises an exception if the receiver is not based on a color space represented by an NSColorSpace object—specifically, colors designated by NSNamedColorSpace and NSPatternColorSpace. If you are unsure about a color object, convert it to an equivalent NSColorSpace-based object before calling this method. Color objects created with color-space names NSCalibratedWhiteColorSpace, NSCalibratedBlackColorSpace, NSCalibratedRGBColorSpace, NSDeviceWhiteColorSpace, NSDeviceBlackColorSpace, NSDeviceRGBColorSpace, NSDeviceCMYKColorSpace, or

`NSCustomColorSpace`—or with the `NSColorSpace` class methods corresponding to these names—are safe to use with this method. See "About Color Spaces" in *Color Programming Topics for Cocoa* for a list of these corresponding methods.

Availability
Available in Mac OS X v10.4 and later.

See Also
– `getComponents:` (page 50)
– `numberOfComponents` (page 56)

Declared In
`NSColor.h`

colorSpaceName

Returns the name of the receiver's color space.

`- (NSString *)colorSpaceName`

Return Value
The name of the color space.

Discussion
This method should be implemented in subclasses of `NSColor`.

Availability
Available in Mac OS X v10.0 and later.

See Also
– `colorUsingColorSpaceName:` (page 47)
– `colorUsingColorSpaceName:device:` (page 47)

Related Sample Code
MyCustomColorPicker

Declared In
`NSColor.h`

colorUsingColorSpace:

Returns a new color object representing the color of the receiver in the specified color space.

`- (NSColor *)colorUsingColorSpace:(NSColorSpace *)space`

Parameters
space
    The color space of the new `NSColor` object.

Return Value
The new `NSColor` object. This method converts the receiver's color to an equivalent one in the new color space. Although the new color might have different component values, it looks the same as the original. Returns `nil` if conversion is not possible.

If the receiver's color space is the same as that specified in `space`, this method returns the same `NSColor` object.

Availability
Available in Mac OS X v10.4 and later.

See Also
+ `colorWithColorSpace:components:count:` (page 21)

Related Sample Code
Quartz 2D Shadings
Quartz Composer QCTV

Declared In
`NSColor.h`

colorUsingColorSpaceName:

Creates and returns an `NSColor` whose color is the same as the receiver's, except that the new `NSColor` is in the specified color space. a

    - (NSColor *)colorUsingColorSpaceName:(NSString *)colorSpace

Parameters
colorSpace
    The name of the color space containing the new `NSColor` object. If *colorSpace* is `nil`, the most appropriate color space is used.

Return Value
The new `NSColor` object or `nil` if the specified conversion cannot be done.

Availability
Available in Mac OS X v10.0 and later.

See Also
– `colorSpaceName` (page 46)

Related Sample Code
CWCocoaComponent
GLChildWindowDemo
MenuItemView
Reminders

Declared In
`NSColor.h`

colorUsingColorSpaceName:device:

Creates and returns an `NSColor` object whose color is the same as the receiver's, except that the new NSColor is in the given color space and is specific to the given device.

```
- (NSColor *)colorUsingColorSpaceName:(NSString *)colorSpace device:(NSDictionary
    *)deviceDescription
```

Parameters

colorSpace

> The name of the color space containing the new `NSColor` object. If *colorSpace* is `nil`, the most appropriate color space is used.

deviceDescription

> The device description. Device descriptions can be obtained from windows, screens, and printers with the `deviceDescription` method.

> If *deviceDescription* is `nil`, the current device (as obtained from the currently lockFocus'ed view's window or, if printing, the current printer) is used.

Return Value

The new NSColor object or `nil` if the specified conversion cannot be done.

Availability

Available in Mac OS X v10.0 and later.

See Also

- `colorSpaceName` (page 46)
- `colorUsingColorSpaceName:` (page 47)

Declared In

`NSColor.h`

colorWithAlphaComponent:

Creates and returns an `NSColor` object that has the same color space and component values as the receiver, but the specified alpha component.

```
- (NSColor *)colorWithAlphaComponent:(CGFloat)alpha
```

Parameters

alpha

> The opacity value of the new `NSColor` object.

Return Value

The new `NSColor` object. If the receiver's color space doesn't include an alpha component, the receiver is returned.

Discussion

A subclass with explicit opacity components should override this method to return a color with the specified alpha.

Availability

Available in Mac OS X v10.0 and later.

See Also

- `alphaComponent` (page 42)
- `blendedColorWithFraction:ofColor:` (page 43)

Related Sample Code

DockTile

FunkyOverlayWindow
ImageMapExample
Sketch-112
SpeedometerView

Declared In
NSColor.h

cyanComponent

Returns the receiver's cyan component.

- (CGFloat)cyanComponent

Return Value
The color object's cyan component.

Discussion
This method works only with objects representing colors in the NSDeviceCMYKColorSpace color space. Sending it to other objects raises an exception.

Availability
Available in Mac OS X v10.0 and later.

See Also
- getCyan:magenta:yellow:black:alpha: (page 50)

Declared In
NSColor.h

drawSwatchInRect:

Draws the current color in the given rectangle.

- (void)drawSwatchInRect:(NSRect)*rect*

Parameters
rect
    The rectangle in which to draw the color.

Discussion
Subclasses adorn the rectangle in some manner to indicate the type of color. This method is invoked by color wells, swatches, and other user interface objects that need to display colors.

Availability
Available in Mac OS X v10.0 and later.

Declared In
NSColor.h

getComponents:

Returns the components of the receiver as an array.

```
- (void)getComponents:(CGFloat *)components
```

Parameters

components

       An array containing the components of the color object as `float` values.

Discussion

You can invoke this method on `NSColor` objects created from custom color spaces to get the individual floating point components, including alpha. Raises an exception if the receiver doesn't have floating-point components. To find out how many components are in the *components* array, send the receiver a `numberOfComponents` (page 56) message.

Availability

Available in Mac OS X v10.4 and later.

See Also

– `colorSpace` (page 45)

Related Sample Code

Quartz Composer QCTV

Declared In

NSColor.h

getCyan:magenta:yellow:black:alpha:

Returns the receiver's CMYK and opacity values.

```
- (void)getCyan:(CGFloat *)cyan magenta:(CGFloat *)magenta yellow:(CGFloat *)yellow
    black:(CGFloat *)black alpha:(CGFloat *)alpha
```

Parameters

cyan

       Upon return, contains the cyan component of the color object.

magenta

       Upon return, contains the magenta component of the color object.

yellow

       Upon return, contains the yellow component of the color object.

black

       Upon return, contains the black component of the color object.

alpha

       Upon return, contains opacity value of the color object.

Discussion

If `NULL` is passed in as an argument, the method doesn't set that value. This method works only with objects representing colors in the `NSDeviceCMYKColorSpace`. Sending it to other objects raises an exception.

Availability

Available in Mac OS X v10.0 and later.

See Also
- alphaComponent (page 42)
- blackComponent (page 42)
- cyanComponent (page 49)
- magentaComponent (page 55)
- yellowComponent (page 60)

Declared In
NSColor.h

getHue:saturation:brightness:alpha:

Returns the receiver's HSB component and opacity values in the respective arguments.

```
- (void)getHue:(CGFloat *)hue saturation:(CGFloat *)saturation brightness:(CGFloat
    *)brightness alpha:(CGFloat *)alpha
```

Parameters

hue
> Upon return, contains the hue component of the color object.

saturation
> Upon return, contains the saturation component of the color object.

brightness
> Upon return, contains the brightness component of the color object.

alpha
> Upon return, contains the opacity value of the color object.

Discussion
If NULL is passed in as an argument, the method doesn't set that value. This method works only with objects representing colors in the NSCalibratedRGBColorSpace or NSDeviceRGBColorSpace color space. Sending it to other objects raises an exception.

Availability
Available in Mac OS X v10.0 and later.

See Also
- alphaComponent (page 42)
- brightnessComponent (page 44)
- hueComponent (page 54)
- saturationComponent (page 57)

Related Sample Code
MenuItemView

Declared In
NSColor.h

getRed:green:blue:alpha:

Returns the receiver's RGB component and opacity values in the respective arguments.

```
- (void)getRed:(CGFloat *)red green:(CGFloat *)green blue:(CGFloat *)blue
    alpha:(CGFloat *)alpha
```

Parameters

red

> Upon return, contains the red component of the color object.

green

> Upon return, contains the green component of the color object.

blue

> Upon return, contains the blue component of the color object.

alpha

> Upon return, contains the opacity value of the color object.

Discussion

If `NULL` is passed in as an argument, the method doesn't set that value. This method works only with objects representing colors in the `NSCalibratedRGBColorSpace` or `NSDeviceRGBColorSpace` color space. Sending it to other objects raises an exception.

Availability

Available in Mac OS X v10.0 and later.

See Also

– alphaComponent (page 42)
– blueComponent (page 43)
– greenComponent (page 53)
– redComponent (page 56)

Related Sample Code

CWCocoaComponent

GLChildWindowDemo

OpenGLCompositorLab

Quartz 2D Shadings

Reminders

Declared In

NSColor.h

getWhite:alpha:

Returns the receiver's grayscale value and alpha values.

```
- (void)getWhite:(CGFloat *)white alpha:(CGFloat *)alpha
```

Parameters

white

> Upon return, contains the grayscale value of the color object.

alpha

> Upon return, contains the opacity value of the color object.

Discussion

If `NULL` is passed in as an argument, the method doesn't set that value. This method works only with objects representing colors in the `NSCalibratedWhiteColorSpace`, `NSCalibratedBlackColorSpace`, `NSDeviceBlackColorSpace`, or `NSDeviceWhiteColorSpace` color space. Sending it to other objects raises an exception.

Availability

Available in Mac OS X v10.0 and later.

See Also

– `alphaComponent` (page 42)

– `whiteComponent` (page 59)

Declared In

`NSColor.h`

greenComponent

Returns the receiver's green component.

```
- (CGFloat)greenComponent
```

Return Value

The color object's green component.

Discussion

This method works only with objects representing colors in the `NSCalibratedRGBColorSpace` or `NSDeviceRGBColorSpace` color space. Sending it to other objects raises an exception.

Availability

Available in Mac OS X v10.0 and later.

See Also

– `getRed:green:blue:alpha:` (page 51)

Related Sample Code

Color Sampler

QTCoreVideo301

Declared In

`NSColor.h`

highlightWithLevel:

Returns an `NSColor` object that represents a blend between the receiver and the highlight color returned by `highlightColor` (page 31).

```
- (NSColor *)highlightWithLevel:(CGFloat)highlightLevel
```

Parameters

highlightLevel

> The amount of the highlight color that is blended with the receiver's color. This should be a number from 0.0 through 1.0. A *highlightLevel* below 0.0 is interpreted as 0.0; a *highlightLevel* above 1.0 is interpreted as 1.0.

Return Value

The new `NSColor` object. Returns `nil` if the colors can't be converted.

Discussion

Invoke this method when you want to brighten the receiving `NSColor` for use in highlights.

Availability

Available in Mac OS X v10.0 and later.

See Also

– `shadowWithLevel:` (page 58)

Declared In

`NSColor.h`

hueComponent

Returns the hue component of the HSB color equivalent to the receiver.

```
- (CGFloat)hueComponent
```

Return Value

The color object's hue component.

Discussion

This method works only with objects representing colors in the `NSCalibratedRGBColorSpace` or `NSDeviceRGBColorSpace` color space. Sending it to other objects raises an exception.

Availability

Available in Mac OS X v10.0 and later.

See Also

– `getHue:saturation:brightness:alpha:` (page 51)

Declared In

`NSColor.h`

localizedCatalogNameComponent

Returns the name of the catalog containing the receiver's name as a localized string.

```
- (NSString *)localizedCatalogNameComponent
```

Return Value

The name of catalog containing the color object's name as a localized string. This string may be displayed in user interface items like color pickers.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ `colorWithCatalogName:colorName:` (page 20)
– `colorNameComponent` (page 45)

Declared In
`NSColor.h`

localizedColorNameComponent

Returns the name of the receiver as a localized string.

```
- (NSString *)localizedColorNameComponent
```

Return Value
The name of color object as a localized string. This string may be displayed in user interface items like color pickers.

Availability
Available in Mac OS X v10.0 and later.

See Also
+ `colorWithCatalogName:colorName:` (page 20)
– `catalogNameComponent` (page 44)
– `colorNameComponent` (page 45)
– `localizedCatalogNameComponent` (page 54)

Declared In
`NSColor.h`

magentaComponent

Returns the receiver's magenta component.

```
- (CGFloat)magentaComponent
```

Return Value
The color object's magenta component.

Discussion
This method works only with objects representing colors in the `NSDeviceCMYKColorSpace` color space. Sending it to other objects raises an exception.

Availability
Available in Mac OS X v10.0 and later.

See Also
– `getCyan:magenta:yellow:black:alpha:` (page 50)

Declared In
`NSColor.h`

numberOfComponents

Returns the number of components in the receiver.

`- (NSInteger)numberOfComponents`

Return Value
The number of components in the color object. The floating-point components counted include alpha. This method raises an exception if the receiver doesn't have floating-point components.

Availability
Available in Mac OS X v10.4 and later.

See Also
- `colorSpace` (page 45)
- `getComponents:` (page 50)

Declared In
`NSColor.h`

patternImage

Returns the image that the receiver is using as a pattern.

`- (NSImage *)patternImage`

Return Value
The image used by the color object. If the receiver doesn't have an image, this method raises an exception.

Availability
Available in Mac OS X v10.0 and later.

Declared In
`NSColor.h`

redComponent

Returns the receiver's red component.

`- (CGFloat)redComponent`

Return Value
The color object's red component.

Discussion
This method works only with objects representing colors in the `NSCalibratedRGBColorSpace` or `NSDeviceRGBColorSpace` color space. Sending it to other objects raises an exception.

Availability
Available in Mac OS X v10.0 and later.

See Also
– getRed:green:blue:alpha: (page 51)

Related Sample Code
Color Sampler
QTCoreVideo301

Declared In
NSColor.h

saturationComponent

Returns the saturation component of the HSB color equivalent to the receiver.

    – (CGFloat)saturationComponent

Return Value
The color object's saturation component.

Discussion
This method works only with objects representing colors in the NSCalibratedRGBColorSpace or NSDeviceRGBColorSpace color space. Sending it to other objects raises an exception.

Availability
Available in Mac OS X v10.0 and later.

See Also
– getHue:saturation:brightness:alpha: (page 51)

Declared In
NSColor.h

set

Sets the color of subsequent drawing to the color that the receiver represents.

    – (void)set

Discussion
This method should be implemented in subclasses.

Availability
Available in Mac OS X v10.0 and later.

Related Sample Code
Dicey
DockTile
FilterDemo
Sketch-112

WhackedTV

Declared In
NSColor.h

setFill

Sets the fill color of subsequent drawing to the receiver's color.

    - (void)setFill

Discussion
This method should be implemented in subclasses.

Availability
Available in Mac OS X v10.3 and later.

See Also
– setStroke (page 58)

Related Sample Code
JSPong

Declared In
NSColor.h

setStroke

Sets the stroke color of subsequent drawing to the receiver's color.

    - (void)setStroke

Discussion
This method should be implemented in subclasses.

Availability
Available in Mac OS X v10.3 and later.

See Also
– setFill (page 58)

Declared In
NSColor.h

shadowWithLevel:

Returns an NSColor object that represents a blend between the receiver and the shadow color returned by
shadowColor (page 39).

    - (NSColor *)shadowWithLevel:(CGFloat)*shadowLevel*

Parameters

shadowLevel

The amount of the shadow color used for the blend. This should be a number from 0.0 through 1.0. A *shadowLevel* below 0.0 is interpreted as 0.0; a *shadowLevel* above 1.0 is interpreted as 1.0.

Return Value

The new NSColor object. Returns nil if the colors can't be converted.

Discussion

Invoke this method when you want to darken the receiving NSColor for use in shadows.

Availability

Available in Mac OS X v10.0 and later.

See Also

– highlightWithLevel: (page 53)

Declared In

NSColor.h

whiteComponent

Returns the receiver's white component.

- (CGFloat)whiteComponent

Return Value

The color object's white component.

Discussion

This method works only with objects representing colors in the NSCalibratedWhiteColorSpace, NSCalibratedBlackColorSpace, NSDeviceBlackColorSpace, or NSDeviceWhiteColorSpace color space. Sending it to other objects raises an exception.

Availability

Available in Mac OS X v10.0 and later.

See Also

– getWhite:alpha: (page 52)

Declared In

NSColor.h

writeToPasteboard:

Writes the receiver's data to the specified pasteboard.

- (void)writeToPasteboard:(NSPasteboard *)*pasteBoard*

Parameters

pasteBoard

The pasteboard to which to write the receiver's color data. If this pasteboard doesn't support color data, the method does nothing.

Availability
Available in Mac OS X v10.0 and later.

See Also
`+ colorFromPasteboard:` (page 17)

Declared In
`NSColor.h`

yellowComponent

Returns the receiver's yellow component.

```
- (CGFloat)yellowComponent
```

Return Value
The color object's yellow component.

Discussion
This method works only with objects representing colors in the `NSDeviceCMYKColorSpace` color space. Sending it to other objects raises an exception.

Availability
Available in Mac OS X v10.0 and later.

See Also
`- getCyan:magenta:yellow:black:alpha:` (page 50)

Declared In
`NSColor.h`

Constants

For definitions of `NSColor` constants, as well as a discussion of their usage, see "About Color Spaces" in *Color Programming Topics for Cocoa*.

AppKit Versions for NSColor Bug Fixes

The version of the AppKit framework containing a specific bug fix.

```
#define NSAppKitVersionNumberWithPatternColorLeakFix 641.0
```

Constants
`NSAppKitVersionNumberWithPatternColorLeakFix`

> The specific version of the AppKit framework that introduced the fix for correctly autoreleasing objects returned by the `colorWithPatternImage:` (page 24) method. Developers should not need to use this constant unless they are writing applications for Mac OS X v10.1 and earlier.
>
> Available in Mac OS X v10.2 and later.
>
> Declared in `NSColor.h`.

Notifications

NSSystemColorsDidChangeNotification

Sent when the system colors have been changed (such as through a system control panel interface).

This notification contains no notification object and no *userInfo* dictionary.

Availability
Available in Mac OS X v10.0 and later.

Declared In
NSColor.h

Document Revision History

This table describes the changes to *NSColor Class Reference*.

Date	Notes
2009-01-06	Added definition for NSAppKitVersionNumberWithPatternColorLeakFix macro.
2008-10-15	Clarified applicability of certain methods to NSColorSpace-based color objects.
2006-05-23	First publication of this content as a separate document.
	First publication of this content as a separate document.

Index

`windowBackgroundColor` **class method** 40
`windowFrameColor` **class method** 41
`windowFrameTextColor` **class method** 41
`writeToPasteboard:` **instance method** 59

Y

`yellowColor` **class method** 41
`yellowComponent` **instance method** 60