

The recent colors are remembered by Igor when it quits and restored when it restarts if you have selected the “Save and restore color palette’s recent colors” checkbox in the Miscellaneous Settings dialog’s Color Settings category.

Color Blending

In places where you can supply a color in RGB format, you can optionally provide a fourth parameter called “alpha”. Alpha, like the R, G and B values, can range from 65535 (100% opaque) down to 0 (100% transparent). Intermediate values provide translucency. For example:

```
Make jack=sin(x/8),sam=cos(x/6); Display jack, sam
ModifyGraph mode=7,hbFill=2
ModifyGraph rgb(jack)=(65535,32768,32768)
ModifyGraph rgb(sam)=(0,0,65535,40000) // Translucent
```

Color blending was added in Igor Pro 7.

Color blending does not work on Windows in the old GDI graphics mode explained under **Graphics Technology on Windows** on page III-506.

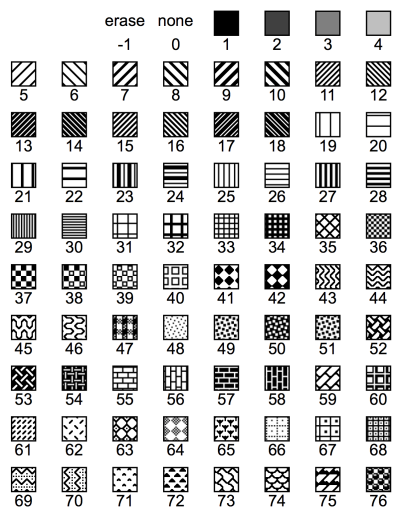
Color blending does not work in graphics exported as EPS. Use PDF instead.

Fill Patterns

Fill patterns can be used in graphs and drawing layers. This table, generated from the ColorsMarkersLine-sPatterns.pxp example experiment, shows the available fill patterns.

The fill pattern codes shown below are appropriate for drawing commands. For graph-related commands, such as ModifyGraph hbFill, make the following adjustments:

- Erase is 1 instead of -1
- Add 1 to all pattern codes greater than 0



Gradient Fills

You can specify gradient fills for drawing elements, graph trace fills, and graph window and plot area background fills. A gradient fill is a gradual change of color.

Programmatically you specify gradients using two keywords, gradient and gradientExtra, when calling the ModifyGraph, ModifyLayout or SetDrawEnv operations.

The syntax for the gradient keyword has two forms. The first form provides overall control of the gradient while the second form controls the color details.

The first form for the gradient keyword is:

```
gradient = 0, 1, 2, or 3
```

0 deletes the gradient entirely.

1 turns on an existing gradient or creates a new one with default values.

2 turns off a gradient but keeps the settings.

3 is used in combination with the ModifyLayout operation and signals that a particular page should not display a gradient even when a layout global gradient is set.

The second form for the gradient keyword is:

```
gradient = {type, x0, y0, x1, y1 [,color0 [, color1 ] ]}
```

type is a bitfield.

Bits 0 through 3 specify the gradient mode. 0 is linear, 1 is radial and other values are reserved.

Bits 4 through 7 specify the coordinate mode. 0 is object rect, 1 is window rect.

To construct a type parameter signifying a radial gradient in window rect mode you could write:

```
Variable gradientType = 1 | (1*16)
```

The following commands illustrate the difference between object rect and window rect mode. Execute these commands and then drag the rectangles around:

```
Display /W=(150,50,474,365)
SetDrawLayer UserFront
SetDrawEnv xcoord=abs,ycoord=abs
SetDrawEnv save
SetDrawEnv fillfgc=(16385,16388,65535),fillbgc=(65535,16385,16385)
SetDrawEnv gradient={16, 0, 0, 1, 1} // Window rect mode
DrawRect 25,38,95,138
SetDrawEnv gradient={1, 0.5, 0.5, 0, 1, (0,65535,0), (65535,0,65535)}
DrawRect 130,150,273,289
ShowTools/A
```

The *x0*, *y0*, *x1*, and *y1* parameters specify normalized locations that define the gradient. (0,0) is the upper left corner of the bounding rectangle while (1,1) is the lower right corner. For a linear gradient, (*x0*, *y0*) defines the start while (*x1*, *y1*) defines the end. Only the slope of the line is used, not the actual extent. For a radial gradient, (*x0*, *y0*) defines the center of a bounding circle while (*x1*, *y1*) is not used at present.

The optional *color0* and *color1* keywords are specified as (r,g,b) or (r,g,b,a) but use of alpha is not fully supported on all platforms (Quartz PDF on Macintosh for example). When omitted or with a value of (0,0,0) the actual color used is a default for the given circumstance. The default values are specified in the individual operation documentation.

The gradientExtra keyword adds an additional color change point. You can add as many change points as desired. The syntax for the gradientExtra keyword is:

```
gradientExtra = {loc, color}
```

loc is in the range 0.0 to 1.0. Values of exactly 0 or 1 replace the original *color0* or *color1* values as specified by the gradient keyword.

color is specified as (r,g,b) or (r,g,b,a) but use of alpha is not fully supported on all platforms (Quartz PDF on Macintosh for example).

The following operations support the gradient and gradientExtra keywords.