Layout

Layout [***flags***] [***objectSpec*** [**,** ***objectSpec***]…][**as** ***titleStr***]

The Layout operation creates a page layout.

Note: The Layout operation is antiquated and can not be used in user-defined functions. For new programming, use the **NewLayout** operation instead.

Parameters

All of the parameters are optional.

Each *objectSpec* parameter identifies a graph, table, textbox or picture to be added to the layout. An object specification can also specify the location and size of the object, whether the object should have a frame or not, whether it should be transparent or opaque, and whether it should be displayed in high fidelity or not. See **Details**.

titleStr is a string expression containing the layout's title. If not specified, Igor will provide one which identifies the objects displayed in the graph.

Flags

/A=(*rows*,*cols*) Specifies rows and columns for tiling or stacking.

/B=(*r*,*g*,*b*[,*a*]) Specifies the background color for the layout. *r*, *g*, *b*, and *a* specify the color and optional opacity as **RGBA Values**. The default is opaque white.

/C=*colorOnScreen* Obsolete. In ancient times, this flag switched the screen display of the layout between black and white and color. It is still accepted but has no effect.

/G=*g* Specifies grout, the spacing between tiled objects. Units are points unless /I, /M, or /R are specified.

/HIDE=*h* Hides (h = 1) or shows (h = 0, default) the window.

/I Specifies that coordinates are in inches. This affects subsequent /G, /W, and *objectSpec* coordinates. Coordinates are relative to the top/left corner of the paper.

/K=*k* Specifies window behavior when the user attempts to close it.

- *k*=0: Normal with dialog (default).
- *k*=1: Kills with no dialog.
- *k*=2: Disables killing.
- *k*=3: Hides the window.

If you use /K=2 or /K=3, you can still kill the window using the **KillWindow** operation.

/M Specifies that coordinates are in centimeters. This affects subsequent /G, /W, and *objectSpec* coordinates. Coordinates are relative to the top/left corner of the paper.

/P=*orientation* *orientation* is either Portrait or Landscape (*e.g.*, `Layout/P= Landscape`). This controls the orientation of the page in the layout. See **Details**.

If you use the /P flag, you should make it the first flag in the Layout operation. This is necessary because the orientation of the page affects the behavior of other flags, such as /T and /G.

/R Specifies that coordinates are in percent. This affects subsequent /G, /W, and *objectSpec* coordinates. For /W, coordinates are as a percent of the main screen. For /G and *objectSpec*, coordinates are relative to the top/left corner of the printing part of the page.

/S Stacks objects.

/T Tiles objects.

/W=(*left*, *top*, *right*, *bottom*)

Gives the layout window a specific location and size on the screen. Coordinates for /W are in points unless /I or /M are specified.

Details

When you create a new page layout window, if preferences are enabled, the page size is determined by the preferred page size as set via the Capture Layout Prefs dialog. If preferences are disabled, as is usually the case when executing a procedure, the page is set to the factory default size.

If you use the /P flag, you should make it the first flag in the Layout operation. This is necessary because the orientation of the page affects the behavior of other flags, such as /T and /G.

The form of an *objectSpec* is:

```
objectName [(objLeft, objTop, objRight, objBottom)][/O=objType][/F=frame]
    [/T=trans][/D=fidelity]
```

objectName can be the name of an existing graph, table or picture. It can also be the name of an object that does not yet exist. In this case it is called a "dummy object".

objectSpec can be specified using a string by using the $ operator, but the entire *objectSpec* must be in the string.

Here are some examples of valid usage:

```
Layout Graph0

Layout/I Graph0(1, 1, 6, 5)/F=1

String s = "Graph0"
Layout/I $s

String s = "Graph0(1, 1, 6, 5)/F=1"
Layout/I $s         // Entire object spec is in string.
```

The object's coordinates are determined as follows:

- If *objectName* is followed by a coordinates specification in (*objLeft*, *objTop*, *objRight*, *objBottom*) form then this sets the object's coordinates. The units for the coordinates are points unless the /I or /M flag was present in which case the units are inches or centimeters respectively.
- If the object coordinates are not specified explicitly but the Layout/S flag was present then the object is stacked. If the Layout/T flag was present then the object is tiled, and if the Layout/A=(*rows*,*cols*) flag is present, tiling is performed using that number of rows and columns.
- If the object's coordinates are not determined by these rules then the object is set to a default size and is stacked.

Each object has a type (graph, table, textbox or picture) determined as follows:

O=*objType* — If the *objectName*/O=*objType* flag is present then it determines the object's type:

objType=1:	Graph.
objType=2:	Table.
objType=8:	Picture.
objType=32:	Textbox.

If there is no /O flag and *objectName* is the name of an existing graph, table or picture, then the object type is graph, table or picture.

If the object's type is not determined by the above rules and *objectName* contains "Table", "PICT", or "TextBox", then the object type is table, picture or textbox.

If the object's type is not specified by any of the above rules, it is taken to be a graph type object.

The remaining flags have the following meanings:

/D=*fidelity* — Controls the drawing of the layout object:

fidelity=0:	Low fidelity display.
fidelity=1:	High fidelity display (default).