To control the color use one of these ModifyContour keywords:

ctabFill	Fills using a color table
cindexFill	Fills using a color index wave
rgbFill	Fills with a specific color

These keywords have the same syntax as `ctabLines`, **cindexLines** and `rgbLines` which control the colors of the contour lines themselves.

To turn on an individual contour level fill, execute:

```
ModifyContour <contour instance name>, fill=0          // Global fill mode off
ModifyGraph usePlusRGB(<contour level trace name>)=1   // Trace fill mode on
ModifyGraph hbFill(<contour level trace name>)=2       // Solid fill for trace
```

For example, in the Contour Demo example experiment, select Macros→Matrix Contour plot to display the Demo Matrix Contour graph. Double-click one of the traces to display the Modify Trace Appearance dialog. Choose Solid for +Fill Type; this automatically checks the Custom Fill checkbox. Select a yellow color from the associated popup menu. This gives the following commands:

```
ModifyGraph usePlusRGB('RealisticData=6')=1
ModifyGraph hbFill('RealisticData=6')=2
ModifyGraph plusRGB('RealisticData=6')=(65535,65532,16385)
```

You can also fill all contour levels, using `ModifyContour fill=1`, and then customize one or more levels using this technique.

You can create a color bar for contour fills using the **ColorScale** operation with the `contourFill` keyword. The syntax is the same as for the ColorScale `contour` keyword.

Solid fills can sometimes fail because Igor can not determine a closed path for a contour line. Be sure to visually inspect the results and turn off fills if they are not correct. The success or failure of a contour fill is highly dependent on the data and is more likely with XYZ data. To see this, choose File→Example Experiments→Sample Graphs→Contour Demo and choose the XYZ Contour Plot from the Macros menu. Turn on Fill Levels and experiment with the number of points and the z-function. Occasionally you may see a warning in the history area saying that a contour level is not closeable. There is not much you can do about this other than trying a different data set or converting your XYZ data to a matrix. Although rare, even matrix data can be sufficiently pathological as to cause the contour fill to fail.

If automatic fills do not work with your data, you can use a background image to provide the fill effect using the WaveMetrics procedure FillBetweenContours. See Image and Contour Plots in the WM Procedures Index for information.

To support fills, Igor needs the boundary trace which it creates and then sets as hidden. When loading an experiment into Igor6, you will encounter an error on the command to hide this trace. You can continue the load by simply commenting out this command in the error dialog.

Removing Contour Traces from a Graph

Removing traces from a contour plot with the RemoveFromGraph operation or the Remove from Graph dialog will work only temporarily. As soon as Igor updates the contour traces, any removed traces may be replaced.

You can prevent this replacement by disabling contour updates with the Modify Contour Appearance dialog. It is better, however, to use the Modify Contour Appearance dialog to control which traces are drawn in the first place.

To permanently remove a particular automatic or manual contour level, you are better off not using manual levels or automatic levels at all. Use the More Contour Levels dialog to explicitly enter all the levels, and enter zero for the number of manual or automatic levels.