To create a label for a given index of a given dimension, use the SetDimLabel operation.

For example:

```
SetDimLabel 1, 0, red, wave2D
```

1 is the dimension number (columns), 0 is the dimension index (column 0) and red is the label.

The function GetDimLabel returns a string containing the name associated with a given dimension and index. For example:

```
Print GetDimLabel(wave2D,1,0)
```

prints “red” into the history area.

The FindDimLabel function returns the index value associated with the given label. It returns the special value -2 if the label is not found. This function is useful in user-defined functions so that you can use a numeric index instead of a dimension label when accessing a wave in a loop. Accessing wave data using a numeric index is much faster than using a dimension label.

In addition to setting the name for individual dimension index values, you can set the name for an entire dimension by using an index value of -1. For example:

```
SetDimLabel 1, -1, ColorComponents, wave2D
```

This sets the label for the columns dimension to “ColorComponents”. This label appears in a table if you display dimension labels.

You can copy dimension labels from one dimension to another or from one wave to another using **CopyDimLabels**.

Dimension labels can contain up to 255 bytes and may contain spaces and other normally illegal characters if you surround the name in single quotes or if you use the $ operator to convert a string expression to a name. For example:

```
wave[%'a name with spaces']
wave[%$"a name with spaces"]
```

Dimension labels have the same characteristics as object names. See **Object Names** on page III-501 for a discussion of object names in general.

Long Dimension Labels

Prior to Igor8, wave dimension labels were limited to 31 bytes. If you create dimension labels of length 31 bytes or fewer, waves saved by Igor8 or later are saved in a format that is compatible with Igor7 or before. If you create a dimension label longer than 31 bytes, that wave is saved in a format that is incompatible with Igor7 and before.

If you attempt to save an Igor binary wave file or an experiment file that has waves with long dimension labels, Igor displays a warning dialog telling you that the experiment will require Igor Pro 8.00 or later. The warning dialog is presented only when you save an Igor binary wave file or experiment interactively, not if you save it programmatically using SaveExperiment. You can suppress the dialog by clicking the "Do not show this message again" checkbox.

Graphing Multidimensional Waves

You can easily view two-dimensional waves as images and as contour plots using Igor’s built-in operations. See Chapter II-15, **Contour Plots**, and Chapter II-16, **Image Plots**, for further information about these types of graphs. You can also create waterfall plots where each column in the matrix wave corresponds to a separate trace in the waterfall plot. For more details, see **Waterfall Plots** on page II-326.