1. **Select wave2 in the Data Browser, drag it to the left3 axis (the upper left axis) in the graph, and pause until New turns blue. Then drag the wave to the Bottom drop target and release the mouse.**

   Wave2 is appended versus the left3 and bottom axes.

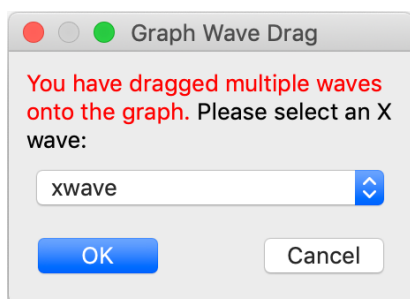2. **Remove the traces from the graph by executing:**

   ```
   RemoveFromGraph wave0, wave1, wave2
   ```

Appending XY Traces by Drag and Drop

To append an XY pair to a graph, you select the X and Y waves in the browser and drag to the graph as described above. When you release the mouse, Igor displays a dialog that allows you to identify one of the waves as the X wave.

1. **Select wave0, wave1, and xwave in the Data Browser, and drag and drop them in the middle of the graph.**

   Igor displays a dialog asking you to select an X wave:

2. **Select xwave from the pop-up menu and click OK.**

   Igor displays wave0 and wave1 versus xwave using the left and bottom axes.

   If you wanted to plot all dragged waves as waveforms rather than as XY pairs, you would choose _calculated_ from the popup menu and click OK.

3. **Remove the traces from the graph by executing:**

   ```
   RemoveFromGraph wave0, wave1
   ```

It is not supported to create multiple XY traces with one drag. For example, if you have waves named wave0 and xwave0 and wave1 and xwave1, you have to do two drags to create two XY pairs. If you have many such XY pairs, use the command line or the advanced mode of the New Graph or Append Traces dialog; see Creating Graphs for details.

Trace Names

Each trace in a graph has a name. Trace names are displayed in graph dialogs and popup menus and used in graphing operations. Some of the operations that take trace names as parameters are **ModifyGraph (traces)**, **RemoveFromGraph**, **ReorderTraces**, **ErrorBars** and **Tag**.

The name of the trace is usually the same as the name of the wave displayed in the trace, but not always. If two traces display the same wave, or waves with the same name from different data folders, **Instance Notation** such as #1and #2, is used to distinguish between the traces. For example:

```
Make wave0 = sin(x/8)

// Create first trace displaying wave0. The trace name is wave0.
Display wave0

// Create second trace displaying of wave0. The trace name is wave0#1.
AppendToGraph wave0
```