GraphPopup Menu

Igor has a contextual menu named "GraphPopup". When you control-click or right-click in a graph away from any trace while in operate mode, you get the GraphPopup menu. You can append menu items to this menu with a GraphPopup menu definition.

For example, the following code adds an "Identify Graph" item to the GraphPopup contextual menu:

```
Menu "GraphPopup"
   "Identify Graph", /Q, IdentifyGraph()
End

Function IdentifyGraph()
   Print WinName(0,1)
End
```

TablePopup Menu

Igor Pro 9.00 and later have a contextual menu named "TablePopup". When you right-click in a table, Igor displays the TablePopup menu. You can append menu items to this menu with a TablePopup menu definition.

For example, the following code adds a "Print Path To Wave" item to the TablePopup contextual menu:

```
Menu "TablePopup", dynamic
   ContextualTableMenuItem(), /Q, PrintPathToSelWave()
End

Function/S ContextualTableMenuItem()
   GetLastUserMenuInfo
   WAVE/Z selWave = $S_firstColumnPath
   if (WaveExists(selWave))
      return "Print Path To Wave"
   endif

   // Here if the user clicked the Point column or an unused column
   return ""         // No menu item is added
End

Function PrintPathToSelWave()
   GetLastUserMenuInfo
   WAVE/Z selWave = $S_firstColumnPath
   if (WaveExists(selWave))
      Print GetWavesDataFolder(selWave,2)
   endif
End
```

DataBrowserObjectsPopup Menu

Igor Pro 9.00 and later have a contextual menu named "DataBrowserObjectsPopup". When you right click the list of objects in the Data Browser, Igor displays the DataBrowserObjectsPopup menu.

You can append menu items to this menu with a DataBrowserObjectsPopup menu definition. If necessary, you can use **GetLastUserMenuInfo** to get information about which menu was selected by the user and **GetBrowserSelection** to determine which objects, if any, are currently selected in the Data Browser.

For example, the following code adds a menu item to the DataBrowserObjectsPopup contextual menu if two numeric waves are selected. The text of the menu item is different depending on whether or not the shift key is pressed when the menu is shown.

```
Menu "DataBrowserObjectsPopup", dynamic
   // This menu item is displayed if the shift key is not pressed
   Display1vs2MenuItemString(0), /Q, DisplayWave1vsWave2(0)

   // This menu item is displayed if the shift key is pressed
   Display1vs2MenuItemString(1), /Q, DisplayWave1vsWave2(1)
End

// If at least two items are selected in the Data Browser object list and the
// first two selected items are numeric waves, this function returns the first
// selected wave via w1 and the second selected wave via w1 unless reverse
// is non-zero in which case the waves are reversed.
// The function result is 1 if the first two selected objects are numeric waves
// and 0 otherwise.
static Function GetWave1AndWave2(WAVE/Z &w1, WAVE/Z &w2, int reverse)
   if (strlen(GetBrowserSelection(-1)) == 0)
      return 0       // Data Browser is not open
   endif

   WAVE/Z w1 = $(GetBrowserSelection(reverse ? 1 : 0))// May be null
   WAVE/Z w2 = $(GetBrowserSelection(reverse ? 0 : 1))// May be null

   if (!WaveExists(w1) || !WaveExists(w2))
      return 0       // Fewer than two waves are selected
   endif

   if (WaveType(w1,1)!=1 || WaveType(w2,1)!=1)
      return 0       // Waves are not numeric
   endif

   return 1
End

Function/S Display1vs2MenuItemString(reverse)
   int reverse    // True (1) if caller wants the reverse menu item string

   int shiftKeyPressed = GetKeyState(0) & 4  // User is asking for reverse?
   if (shiftKeyPressed && !reverse)
      // User is asking for reverse so hide unreversed menu item
      return ""
   endif
   if (!shiftKeyPressed && reverse)
      // User is not asking for reverse so hide reversed menu item
      return ""
   endif

   WAVE/Z w1, w2
   int twoNumericWavesSelected = GetWave1AndWave2(w1, w2, reverse)
   if (!twoNumericWavesSelected)
      return ""
   endif

   String menuText
   sprintf menuText, "Display %s vs %s", NameOfWave(w1), NameOfWave(w2)
   return menuText
End

// If reverse is false, execute Display w1 vs w2
// If reverse is true, execute Display w2 vs w1
Function DisplayWave1vsWave2(int reverse)
   WAVE/Z w1, w2
```