

WMSetVariableAction

```
Structure WMPopupAction
char ctrlName[32]           // Control name
char win[200]               // Host window or subwindow name
STRUCT Rect winRect         // Local coordinates of host window
STRUCT Rect ctrlRect        // Enclosing rectangle of the control
STRUCT Point mouseLoc       // Mouse location
Int32 eventCode             // See details below
Int32 eventMod              // See Control Structure eventMod Field on page III-438
String userData             // Primary unnamed user data
Int32 blockReentry         // Obsolete, see Control Structure blockReentry Field on page
III-439
Int32 popNum                // Item number currently selected or hovered over (1-based)
char popStr[MAXCMDLEN]      // Contents of current popup item or item hovered over
EndStructure
```

The constants used to specify the size of structure char arrays are internal to Igor Pro and may change.

WMPopupAction eventCode Field

Your action function should test the eventCode field and respond only to documented eventCode values because other event codes may be added in the future.

The event code passed to the pop-up menu action procedure has the following meaning:

Event Code	Event
-3	Control received keyboard focus (Igor8 or later)
-2	Control received keyboard focus (Igor8 or later)
-1	Control being killed
2	Mouse up
3	Hovering - sent when the user highlights a menu item by moving the mouse cursor over it but hasn't selected it
4	Dismissed - sent when the user closes the menu without making a selection. This is primarily of use in conjunction with the hover event; it allows you to undo any changes made during a hover event when the menu is dismissed. This event code was added in Igor Pro 9.00.

WMSetVariableAction

This structure is passed to action procedures for SetVariable controls created using the SetVariable operation.

```
Structure WMSetVariableAction
char ctrlName[32]           // Control name
char win[200]               // Host window or subwindow name
STRUCT Rect winRect         // Local coordinates of host window
STRUCT Rect ctrlRect        // Enclosing rectangle of the control
STRUCT Point mouseLoc       // Mouse location
Int32 eventCode             // See details below
Int32 eventMod              // See Control Structure eventMod Field on page III-438
String userData             // Primary unnamed user data
Int32 blockReentry         // Obsolete, see Control Structure blockReentry Field on page
III-439
Int32 isStr                 // TRUE for a string variable
Variable dval               // Numeric value of variable
char sval[MAXCMDLEN]        // Value of variable as a string
char vName[MAX_OBJ_NAME+2 + (MAXDIMS * (MAX_OBJ_NAME+5)) + 1]
WAVE svWave                 // Valid if using wave
Int32 rowIndex              // Row index for a wave if rowLabel is empty
char rowLabel[MAX_OBJ_NAME+1] // Wave row dimension label
Int32 colIndex              // Column index for a wave if colLabel is empty
char colLabel[MAX_OBJ_NAME+1] // Wave column dimension label
Int32 layerIndex            // Layer index for a wave if layerLabel is empty
char layerLabel[MAX_OBJ_NAME+1] // Wave layer label
Int32 chunkIndex            // Chunk index for a wave if chunkLabel is empty
char chunkLabel[MAX_OBJ_NAME+1] // Wave chunk label
Int32 mousePart             // Part of the control where mouse down occurred
EndStructure
```

The constants used to specify the size of structure char arrays are internal to Igor Pro and may change.