```
/*-------------------------------------------------------------------------*
 * File Name: ManipulateGraphAxes.c                                        *
 * Creation: ER, 02/14/05                                                  *
 * Purpose: Programming Example                                            *
 * Copyright (c) OriginLab Corp.2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010  *
 * All Rights Reserved                                                     *
 *                                                                         *
 * Modification Log:                                                       *
 *-------------------------------------------------------------------------*/

#include <Origin.h>

////////////////////////////////////////////////////////////////////////////
// This example shows how to change various properties of graph axes.
// NOTE: It is assumed that a graph layer is active.
//
void manipulate_graph_axes()
{
    // Declare graph layer using active layer
    GraphLayer gly = Project.ActiveLayer();
    if( !gly )
    {
        out_str("Active layer is not a graph!");
        return;
    }

    // Simple manipulation of scale property:
    //
    // Point to X axis of active layer
    Axis xAx = gly.XAxis;
    // Get Scale properties and output to script window
    // Can then see from output what is available
    Tree trFormat;
    trFormat = gly.XAxis.Scale;
    out_str("Scale Properties of X Axis:");
    out_tree(trFormat);
    // Change scale type, and from and to
    xAx.Scale.Type.nVal = 1;    // log scale
    xAx.Scale.From.dVal = 1;
    xAx.Scale.To.dVal = 10000;
    // There is one more property of axis that is directly accesible:
    trFormat = gly.XAxis.Additional;
    out_str("Additional property of X Axis:");
    out_tree(trFormat);
    xAx.Additional.OppositeLine.nVal = 1;   // turn on opposite line

    // Now to access all other properties of the axes, the following needs to be done:
    // Look at the table towards the bottom and decide what property to access.
    // Then get the properties into a tree and output to script window to see
    // what is available. Then set the values as desired.
    //
    // Example1:
    // Major grid properties:
    AxisObject aoX;
    aoX = gly.XAxis.AxisObjects(AXISOBJPOS_MAJOR_GRID);
    trFormat =  aoX.VerticalMajorGrids;
    out_str("Vertical Grid Properties of X Axis:");
    out_tree(trFormat);
    // Turn on vertical major grid and set color to red
    aoX.VerticalMajorGrids.Show.nVal = 1;
    aoX.VerticalMajorGrids.Color.nVal = 1;
    //
    // Example 2:
    // Left axis ticks properties:
    AxisObject aoY;
    aoY = gly.YAxis.AxisObjects(AXISOBJPOS_AXIS_FIRST);
    trFormat =  aoY.LeftTicks;
    out_str("Left Ticks Properties of Y Axis:");
    out_tree(trFormat);
    // Set width and length of ticks to different value
    aoY.LeftTicks.Width.dVal = 4.5;
    aoY.LEftTicks.Length.nVal = 12;
}

/*
Table showing what property is available for X, Y and Z axes:

What Property?  What object to use?         X Axis              Y Axis                  Z Axis
```

```
--------------  -------------------         ------              ------                  -------
Minor grids:    AXISOBJPOS_MINOR_GRID       VerticalMinorGrids  HorizontalMinorGrids    MinorGrids
Major grids:    AXISOBJPOS_MAJOR_GRID       VerticalMajorGrids  HorizontalMajorGrids    MajorGrids
1st label:      AXISOBJPOS_LABEL_FIRST      BottomLabels        LeftLabels              FrontLabels
2nd label:      AXISOBJPOS_LABEL_SECOND     TopLabels           RightLabels             BackLabels
1st ticks:      AXISOBJPOS_AXIS_FIRST       BottomTicks         LeftTicks               FrontTicks
2nd ticks:      AXISOBJPOS_AXIS_SECOND      TopTicks            RightTicks              BackTicks
*/
////////////////////////////////////////////////////////////////////////////////
```