List Manager Demo

```
// An example illustrating the use of the List Manager
// Clicking a list element causes it to be highlighted
// Double-clicking a list element causes it to beep.

// Assuming inclusion of <MacHeaders>

#include <stdio.h>
#include <string.h>

void ToolBoxInit(void);

void ToolBoxInit()
{
    InitGraf(&thePort);
    InitFonts();
    InitWindows();
    TEInit();
    InitDialogs(nil);
    InitCursor();
}

main()
{
    ListHandle      myList;
    WindowPtr       myWindow;
    Rect            tempR, dataBounds;
    Point           cSize;
    short           i, dummy;
    Str255          s;
    Boolean         done;
    EventRecord     theEvent;
    WindowPtr       whichWindow;
    long            windSize;
    short           thePart;
    Rect            sizeRect;
    GrafPtr         oldPort;
    Boolean         beep;

    ToolBoxInit();
    SetRect(&tempR, 20, 50, 400, 300);
    myWindow = NewWindow(nil, &tempR, "\pList Mgr Demo",
                                        true, documentProc, (WindowPtr) -1,
                                        true, 0);
    SetPort((GrafPtr)myWindow);
    SetRect(&dataBounds, 0, 0, 1, 0);
    SetPt(&cSize, 0, 0);
    SetRect (&tempR, 0, 0, 380 -15, 250 -15);
    myList = LNew(&tempR, &dataBounds, cSize, 0,
                                        (WindowPtr)myWindow,true, true, true,
                                        true);
    HLock((Handle)myList);
    cSize = (*myList)->cellSize;   // Save cellSize so we can use
                                   // it later to resize the cells
    InitCursor();
```

```
DrawGrowIcon ((WindowPtr) myWindow);
SetRect(&tempR, tempR.left - 1, tempR.top - 1,
                                    tempR.right + 16, tempR.bottom + 1);
for (i = 0; i < 20; ++i) {
        dummy = LAddRow(1, i, myList);
        SetPt(&cSize, 0, i);
        sprintf((char *)s, "Row #%d", i);
        LAddToCell(s, strlen((char *) s), cSize, myList);
        LDraw(cSize, myList);
}

done = false;
SetRect (&sizeRect, 50, 50,
                screenBits.bounds.bottom - screenBits.bounds.top,
                screenBits.bounds.right - screenBits.bounds.left);
done = FALSE;
while (!done)
        if ( WaitNextEvent (everyEvent, &theEvent, 0, nil ) ) {
                switch (theEvent.what) {
                case mouseDown:
                        thePart = FindWindow (theEvent.where,
                                &whichWindow);

                        switch (thePart)
                        {
                        case inContent:
                                GlobalToLocal(&theEvent.where);
                                beep = LClick(theEvent.where,
                                        theEvent.modifiers, myList);
                                if (beep)
                                        SysBeep (10);
                                break;
                        case inGrow:
                                windSize = GrowWindow (whichWindow,
                                        theEvent.where, &sizeRect);
                                if (windSize)
                                {
                                        GetPort (&oldPort);
                                        SetPort (whichWindow);
                                        EraseRect (&whichWindow->portRect);

                                        SizeWindow (whichWindow,
                                                LoWord (windSize),
                                                HiWord(windSize), true);

                                        LSize (LoWord(windSize)-15,
                                                HiWord (windSize)-15, myList);

                                        cSize.h = LoWord(windSize)-15;
                                        LCellSize (cSize, myList);
                                        DrawGrowIcon((GrafPtr)myWindow);
                                        InvalRect (&whichWindow->portRect);
                                        SetPort (oldPort);
                                }
                                break;
                        case inGoAway:
```

```
                        done = TRUE;
                    }
                    break;
                case updateEvt:
                    if ((WindowPtr) theEvent.message == myWindow) {
                        BeginUpdate ((WindowPtr) theEvent.message);
                        DrawGrowIcon ((WindowPtr) myWindow);
                        for (i = 0; i < 20; ++i) {
                            SetPt(&cSize, 0, i);
                            LDraw(cSize, myList);
                        }
                        EndUpdate ((WindowPtr) theEvent.message);
                    }
                    break;
                }
            }
    HUnlock((Handle) myList);
    LDispose(myList);
    DisposeWindow((WindowPtr)myWindow);
}
```