LActivate Activate or deactivate a list (after activate event)

#include <Lists.h> **List Manager Package**

void **LActivate(***activateIt*, *theList* **)**;
Boolean *activateIt* ; TRUE=activate; FALSE=deactivate
ListHandle *theList* ; handle leading to a ListRec

Call **LActivate** in response to an activate event for a list's enclosing window. It highlights or unhighlights any cells that are currently selected and shows or hides the scroll bars (if any).

activateIt specifies whether to activate or deactivate the list. It is one of:

FALSE Deactivate the list.

TRUE Activate the list.

theList is a handle leading to a variable-length ListRec structure. It is a value previously obtained via **LNew**.

Returns: none

Notes: The **List Manager** does NOT take care of the size box (grow icon) of lists that can be sized, so you must draw or erase it yourself. Here's a fragment of a main event loop that maintains a growable list:

```
WindowPtr    listWindow;          // assume this already exists
EventRecord  theEvent;
ListHandle theList;

// in event loop

if(WaitNextEvent(everyEvent, &theEvent, 0, nil)) {

    if ( theEvent.what == activateEvt ) {
        if ( theEvent.message == (long)listWindow ) {
            if ( theEvent.modifiers & activeFlag )
                                        // it's an activate request
                LActivate( TRUE, theList );// U get bonus for ternary op
            else
                LActivate( FALSE, theList );    // it's a deactivate request
            DrawGrowIcon( listWindow );     // do this in either case
        }
    }
}
```

You may not need or want to deactivate a list displayed in a modeless dialog box. Calling **LActivate(**FALSE,...) causes the scroll bars to go away entirely.