LAddColumn Insert column(s) of empty cells into a list

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

short	**LAddColumn(***count*, *clmNum*, *theList* **);**	
short	*count* ;	how many columns to insert
short	*clmNum* ;	where to start inserting
ListHandle	*theList* ;	handle leading to a ListRec
	returns	column number of the first inserted column

LAddColumn inserts one or more columns of empty cells into a list. If drawing is on, the list display and the horizontal scroll bar (if any) are updated.

count specifies the number of columns to insert.

clmNum specifies where to start inserting columns. Columns are inserted *before* this column. For instance, if *clmNum*=3 and *count*=1, columns 3...*n* are renumbered as columns 4...*n*+1. Thus, the cell that used to be called (3,0) is now called (4,0), and so forth.

If *clmNum* > ListRec.dataBounds.right (i.e., greater than the current width), then exactly *count* columns are added to the rightmost side of the list. The column where they were actually added is returned.

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

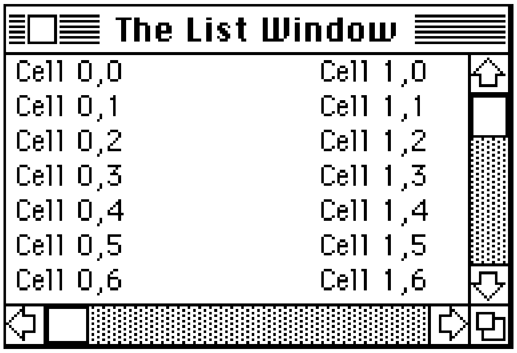
Returns: a short; the column number of the first column inserted. When inserting within the array bounds, this simply returns *clmNum*. But, if you attempt to insert beyond the current bounds, the return value is the current horizontal size of the list (i.e., ListRec.dataBounds.right).

Notes: **LAddColumn** increases the size of the ListRec structure by (*count* * ListRec.dataBounds.bottom) * 2 bytes. ListRec.dataBounds.right is increased by *count.*

For instance, after:

```
LAddColumn( 1,1, theList );        /* insert 1 column at column 1 */
```

The list shown in the **LNew** example changes to look like:



Cell 0,0	Cell 1,0
Cell 0,1	Cell 1,1
Cell 0,2	Cell 1,2
Cell 0,3	Cell 1,3
Cell 0,4	Cell 1,4
Cell 0,5	Cell 1,5
Cell 0,6	Cell 1,6

Note that if there are no rows or columns (as when *rDataBnds* is empty when you call **LNew**), you must insert at least one row or column (via **LAddRow** or **LAddColumn**) before starting to store cell data via **LSetCell**.