DisableItem   Dim a menu or a menu item; make non-selectable

#include <Menus.h>   **Menu Manager**

void	**DisableItem(**_theMenu_, _whichItem_ **);**	
MenuHandle	_theMenu_ ;	handle of menu to affect
short	_whichItem_ ;	item ID of item to disable; 0=entire menu

DisableItem dims a menu title or an individual item in a menu. When disabled, the item (or menu) is not highlighted when the cursor moves over it and cannot be selected.

theMenu is a handle leading to a variable-length MenuInfo structure. It is a value obtained via **NewMenu** or **GetMenu**. It identifies the menu containing the item to disable, or if _whichItem_ = 0, the menu to disable.

whichItem identifies which item to disable or specifies your intention to disable an entire menu. When _whichItem_=0, the menu title is dimmed and all items in that menu are disabled. If it is a valid item ID, it specifies an item in _theMenu_. Items are numbered sequentially with the topmost item having an ID of 1; the last item ID can be obtained via **CountMItems**

Returns: none

───

Notes: Use **DisableItem** on all menu items that are not valid in the current context; e.g., disable "Paste" when there is nothing in the paste buffer.

Disabling an item also excludes it from the search performed by **MenuKey**. Use **EnableItem** when you want the item to be selectable.

When you disable or re-enable an entire menu (_whichItem_ =0), you should call **DrawMenuBar** to display the changed menu title.

This works by clearing bit number _whichItem_ in the enableFlags field of the relevant MenuInfo structure. For instance, the following calls are functionally equivalent:

DisableItem(myMenu, 2);
(*myMenu)-> enableFlags &= ~4;   /* clear bit 2 */

All menus and items are implicitly enabled when inserted into the menu unless explicitly disabled by including the "❨" metacharacter in the item text. Items with IDs higher than 31 are always enabled, since there is no "enable flag" to control them.

───