

SetItemCmd

Assign a command-key to a menu item; make submenu

#include <Menus.h>

Menu Manager

```

void      SetItemCmd(theMenu, whichItem, cmdChar);
MenuHandle theMenu ;      handle of menu containing item to modify
short whichItem ;      ID of an item in theMenu
char cmdChar ;      command character for menu; 0x1B=submenu

```

SetItemCmd assigns a command-key equivalent to a menu item (so that subsequent calls to **MenuKey** can return *whichItem*). It also provides a way to convert a normal menu item into a hierarchical submenu title.

theMenu is a handle leading to a variable-length MenuInfo structure. It is a value obtained via **NewMenu** or **GetMenu** and it identifies the menu containing the item to modify.

whichItem identifies which item in *theMenu* to modify. Items are numbered sequentially with the topmost item having an ID of 1.

cmdChar is a character to be used as the command-key equivalent for menu item *whichItem* (i.e., a value of 'B' or 'b' will cause a call to **MenuKey** to return *whichItem* after the user presses **B** or **b**).

If *cmdChar* is hMenuCmd (0x1B), *whichItem* becomes submenu title and you should call **SetItemMark** to identify which submenu should be attached to *whichItem*.

Returns: none

Notes: **SetItemCmd** is new with the 256K ROMs. It provides a way to change which command key is associated with a menu item or to convert a simple menu item into a hierarchical submenu title.

Mac User Guidelines recommend that you never change command-key equivalents unless a user has specifically requested you to do so. Furthermore, the normal way to set up a command key for an item is to use the '/' metacharacter in the initial call to **InsMenuitem** or **AppendMenu** (or when creating the menu via a resource tool).

This call would be more often used to toggle an item to and from a submenu. For instance, a menu item named **Close MyFile**, seen when only one window is open, might be changed to **Close Window** (a submenu title) when two or more windows are open.

```
MenuHandle windowsSubMenu, otherMenu;
```

```
/* after user has opened a second window */
/* create submenu with two entries ... */
```

```
windowsSubMenu = NewMenu( SMID_CLWIND, "\pClose Window" );
AppendMenu( windowsSubMenu, "\pMyFile" );
```

AppendMenu(windowsSubMenu, "\pMy2ndFile");

SetItem(otherMenu, ITM_CLWIND, "\pClose Window");

SetItemCmd(otherMenu, ITM_CLWIND, hMenuCmd);

SetItemMark(otherMenu, ITM_CLWIND, SMID_CLWIND);