20. FREE MENUS

Free Menu is a library package that is even more flexible than the regular menu package. It allows you to create menus with different types of items in them, and formats them as you require. Free menus are particularly useful when you want a "fill in the form" type interaction with the user.

Each menu item is described with a list of properties and values. The following example will give you an idea of the structure of the description list, and some of your options. The most commonly used properties, and each type of menu item will be described in the Parts of a Free Menu Item and Types of Free Menu Items section below.

Free Menu Example

Free menus can be created and formatted automatically! It is done with the function FM.FORMATMENU. This function takes one argument, a description of the menu. The description is a list of lists; each internal list describes one row of the free menu. A free menu row can have more than one item in it, so there are really lists of lists of lists! It really isn't hard, though, as you can see from the following example:

```
(SETQ ExampleMenu
    (FM.FORMATMENU
        '(((TYPE TITLE LABEL TitlesDoNothing)
           TYPE 3STATE LABEL Example3State))
          ((TYPE EDITSTART LABEL PressToStartEditing
                           ITEMS (EDITEM))
           (TYPE EDIT ID EDITEM LABEL ""))
        (WINDOWPROPS TITLE "Example Does Nothing"))))
```

The first row has two items in it: one is a TITLE, and the second is a 3STATE item. The second row also has two items. The second, the EDIT item, is invisible, because its label is an empty string. The caret will appear for editing, however, if the EDITSTART item is chosen. Windowprops can appear as part of the description of the menu, because a menu is, affer all, just a special window. You can specify not only the title with WINDOWPROPS, but also the position of the free menu, using the "left" and "bottom" properties, and the width of the border in pixels, with the "border" property. Evaluating this expression will return a window. You can see the menu by using the function OPENW. The following example illustrates this:

Figure 20.1. Example Free Menu

The next example shows you what the menu looks like after the EDITSTART item, PressToStartEditing, has been chosen.

Figure 20.2. Free menu after EDITSTART Item Chosen

The following example shows the menu with the 3STATE item in its T state, with the item highlighted. (In the previous bitmaps, it was in its neutral state.)

.

Figure 20.3. Free menu with 3STATE Item in its T State

Finally, Figure 20.4 shows the 3STATE item in its NIL state, with a diagonal line through the item

Figure 20.4 Free menu with the 3STATE item in its NIL State

If you would like to specify the layout yourself, you can do that too. See the *Lisp Library Packages Manual* for more information.

Parts of a Free Menu Item

There are eight different types of items that you can use in a free menu. No matter what type, the menu item is easily described by a list of properties, and values. Some of the properties you will use most often are listed below:

LABEL	Required for every type of menu item. It is the atom, string, or bitmap that appears as a menu selection.
TYPE	One of eight types of menu items. Each of these are described in the section below.
MESSAGE	The message that appears in the prompt window if a mouse button is held down over the item.
ID	An item's unique identifier. An ID is needed for certain types of menu items.
ITEMS	Used to list a series of choices for an NCHOOSE item, and to list the ID's of the editable items for an EDITSTART item.
SELECTEDFN	The name of the function to be called if the item is chosen.

Types of Free Menu Items

Each type of menu item is described in the following list, including an example description list for each one.

MOMENTARY — This is the familiar sort of menu item. When it is selected, the function stored with it is called. A description for the function that creates and formats the menu looks like this:

```
(TYPE MOMENTARY
    LABEL Blink-N-Ring
    MESSAGE "Blinks the screen and rings bells"
    SELECTEDFN RINGBELLS)
```

TOGGLE — This menu item has two states, T and NIL. The default state is NIL, but choosing the item toggles its state. The following is an example description list, without code for the SELECTEDFN function, for this type of item:

```
(TYPE TOGGLE
    LABEL DwimDisable
    SELECTEDFN ChangeDwimState)
```

3STATE — This type of menu item has three states, NEUTRAL, T, and NIL. NEUTRAL is the default state. T is shown by highlighting the item, and NIL is shown with diagonal lines. The following is an example description list, without code for the SELECTEDFN function, for this type of item:

```
(TYPE 3STATE
    LABEL CorrectProgramAllOrNoSpelling
    SELECTEDFN ToggleSpellingCorrection)
```

TITLE — This menu item appears on the menu as dummy text. It does nothing when chosen. An example of its description:

```
(TYPE TITLE LABEL "Choices:")
```

NWAY — A group of items, nnly one of which can be chosen at a time. The items in the NWAY group should all have an ID field, and the ID's should be the same. For example, to set up a menu that would allow the user to choose between Helvetica, Gacha, Modern, and Classic fonts, the descriptions might look like this (once again, without the code for the SELECTEDFN):

```
(TYPE NWAY ID FONTCHOICE
    LABEL Helvetica
    SELECTEDFN ChangeFont)
(TYPE NWAY ID FONTCHOICE
    LABEL Gacha
    SELECTEDFN ChangeFont)
(TYPE NWAY ID FONTCHOICE)
    LABEL Modern
    SELECTEDFN ChangeFont)
(TYPE NWAY ID FONTCHOICE
    LABEL Classic
    SELECTEDFN Changefont)
```

NCHOOSE — This type of menu item is like NWAY except that the choices are given to the user in a submenu. The list to specify an NCHOOSE menu item that is analogous to the NWAY item above might look like this:

```
(TYPE NCHOOSE
    LABEL FontChoices
    ITEMS Helvetica Gacha Modern Classic)
    SELECTDFN Changefont)
```

EDITSTART — When this type of menu itein is chosen, it activates another type of item, an EDIT item. The EDIT item or items associated with an EDITSTART item have their lD's listed on the EDITSTART's ITEMS property. An example description list is:

```
(TYPE EDITSTART LABEL "Function to add?" ITEMS (Fn))
```

EDIT — This type of menu item can actually be edited by you. It is often associated with an EDITSTART item (see above), but the caret that prompts for input will also appear if the item itself is chosen. An EDIT item follows the same editing conventions as editing in Executive Window:

Add characters by typing them at the caret.

Move the caret by pointing the mouse at the new position, and clicking the left button.

Delete characters from the caret to the mouse by pressing the right button of the mouse. Delete a character behind the caret by pressing the backspace key.

Stop editing by typing a carriage return, a Control-X, or by choosing another item from the menu.

An example description list for this type of item is:

```
(TYPE EDIT ID Fn LABEL **)
```