

# **Environment Control Panel Applet**

(c)1993 by Mark Gamber

**[About This Program](#)**

**[Applet Display](#)**

**[Adding New Settings](#)**

**[Removing Existing Settings](#)**

**[Modifying Current Settings](#)**

## About Environment Control Panel Applet

An **Environment** is a set of variables used to govern the behavior of applications running within a system. The most common environment variable is the **PATH** string, which defines where the shell program looks for files if not in the current directory. Other environment strings exist for specific applications, such as the Windows SDK's "**Include**" and "**Lib**" environment settings.

When a user is logged into Windows NT, two environments exist for the session. The **System Environment** defines the environment settings for the machine and is the same for all users that may log into the machine. **User Environment** differs from user to user, so settings made to your User Environment will not affect others able to log into the same machine.

The settings available from this applet can be found within the Registry but this applet was written to make it easier to change the environment settings. The environment type in use is selected by checking either the "System Environment" or "User Environment" box above each listbox, selecting an item and performing an action on the item or listbox. Changes are made immediately to the Registry.

This program was written by Mark Gamber and is Public Domain.

## Adding New Settings

Press **Alt-A** or press the "**Add New**" button to display a dialog box to receive the new environment variable name and value. Enter a valid name and value and press **Enter** or "**OK**" to accept the values, or **ESC** or **Cancel** to abort the entry without change.

Upon accepting the new values, they are added to the environment selected by the radio buttons above the listboxes. If an error occurs, it is displayed and the operation is aborted.

See Also: [Removing Existing Settings](#), [Modifying Current Settings](#)

## Removing Existing Settings

Select the environment type by checking one of the radio buttons above each listbox and select an item within the listbox to delete. Press **Alt-D** or "**Delete**" to remove the item from the selected listbox. If an error occurs, it is displayed and the operation is aborted.

See Also: [Adding New Settings](#), [Modifying Current Settings](#)

## Modifying Current Settings

Select the environment type by checking one of the radio buttons above each listbox and select an item within the listbox to modify. Press **Alt-M** or "**Modify**" to display the item in "Modify" mode. A text entry field is displayed directly over the selected listbox item with the current contents, if any, displayed within the entry field. Modify the contents to suit and press **Enter** or "**OK**" to accept the field contents or **ESC** or **Cancel** to abort without change.

The Registry is written to reflect the change immediately upon accepting the value in the entry field. If an error occurs, it is displayed and the process is aborted. In any case, the text entry field disappears and the listbox resumes normal control.

See Also: [Adding New Settings](#), [Removing Existing Settings](#)

## Applet Display

The main display of the Environment Applet is broken into two sections, one radio button and listbox for System Environment variables and another for User Environment variables. One of the two radio buttons is always checked. The checked button represents that the listbox below it has been selected for button operations. That is, selecting the "User Environment" radio button and adding a new variable, deleting a variable or modifying a variable will affect the user environment and cause a change in that listbox.

Listboxes may be selected by choosing a radio button with a pointer or by the keyboard and using the up and down arrow keys to move into the listbox below and move among the listbox items. When a listbox is clicked on directly using a pointer, the radio button selection is changed automatically to reflect the listbox change.

