August, 1996

Advisor Answers

Visual FoxPro 3.0

Q: My question is related to the KeyPress event (for a text box or other control). The parameter nKeyCode seems to be passed by value unlike Visual Basic 4.0 where it is passed by reference.

Passing the parameter by reference allows you to get control of the value of this parameter and change it through code in the KeyPress event before it is passed in turn to the textbox. You can discard some characters or change others and therefore get full control over user input.

SET UDFPARMS works only for user defined functions and has no effect on event triggered methods. Is there any solution to this problem ?

–Maurice Zaidan (via the Internet)

A: Visual FoxPro's way of handling this sort of thing is different than Visual Basic's and is more general. Every method and most events have some default behavior that occurs when they execute. For example, the default behavior of KeyPress is to put the keystroke into the keyboard buffer where it can be processed. The default behavior of OpenTables is to open the tables in the data environment, and so forth. The default behavior is always the last thing that the method or event does - any custom code is processed first.

VFP provides a command which prevents the default behavior. The keyword is NODEFAULT and it's available only in method code. Issuing NODEFAULT in KeyPress prevents the keystroke from being placed in the keyboard buffer.

But what if you want to change the keystroke, not just prevent it? Use the KEYBOARD command. For example, suppose you want to ensure that every key is upper case (of course, a Format of ! does this for controls with a Format property). You could put the following code in KeyPress:

```
IF BETWEEN(CHR(nKeyCode),"a","z")  && little more readable?
   NODEFAULT
   KEYBOARD CHR(nKeyCode-32)     && move to upper case
ENDIF
```

Keep NODEFAULT in mind anytime you want to change the native behavior of a method.

–Tamar