

June, 1998

## Advisor Answers

Visual FoxPro 5.0 and 3.0 and FoxPro 2.x

Visual FoxPro 5.0

Q: I haven't been able to find information on turning off the display of current table, record number, and so forth on the bottom of the main Visual FoxPro 5.0 window. I want to be able to see the status of NumLock, CapsLock and Insert.

–Ron Booth (via Advisor.Com)

A: Prior to VFP 5, there was no easy way to get what you want. But, with the exposure of the VFP application object, this one is actually a one-liner. The application object has a StatusBar property that determines what appears in the status bar. To clear the status bar, use:

```
_VFP.StatusBar = ""
```

If you'd rather put a message there, use something like:

```
_VFP.StatusBar = "Welcome to Ron's Cool Application!"
```

The toggles for the NumLock, CapsLock and Insert buttons still work, whether the rest of the status bar is blank or contains a custom message.

There are a few problems with manipulating the StatusBar property. First, I've found no command to restore the status bar to its default VFP behavior. However, this isn't really the biggest problem.

In addition to showing the contents of the application's StatusBar property, the status bar also shows the contents of the StatusBarText property of controls in the forms you run. (This is a very nice way to give users a little help that doesn't get in the way.) However, as soon as focus leaves a control that has StatusBarText and goes to a control for which that property is empty, the status bar returns to showing table information. Similarly, when you close a form in which a control has displayed StatusBarText, the status bar shows table information. Also, a number of other operations (like opening a table) return the status bar to its default behavior.

You can work around this issue by resetting `_VFP.StatusBar` to your desired message regularly. If all the controls in all your forms have `StatusBarText`, just reset `_VFP.StatusBar` in the Destroy event of each form. If some controls don't have `StatusBarText` (though I tend to think either all should or none should, from an interface point of view), you'll have to reset `_VFP.StatusBar` more often, perhaps in each control's `LostFocus`.

Of course, having to do that much work (though you can do it in your base classes) raises the question of whether it's worth it. In many applications, showing the user information about the table in use isn't really a problem. In applications where showing this information is a problem, then it's worth the trouble.

There is one other alternative. You can remove the built-in status bar entirely and use an ActiveX status bar instead. For ideas on how to go about that and the issues this approach raises, see the ActiveX Control Counter column in the March '98 FoxPro Advisor.

-Tamar