

Ctrl-C, Ctrl-X and Ctrl-V in textbox with Visual Basic.Net

Are you programming an **MDI-application**, and are you challenged by the fact that your MDI-childs won't process Ctrl-C, Ctrl-X and Ctrl-V in textboxes?

I did, and my first thought was, that Visual Basic.Net didn't support handling of **Ctrl-C** etc. in a textbox.

To test this, I made a small test-program (a simple SDI application with one form and one textbox). I discovered that Visual Basic.Net *does* handle these shortcuts properly in textboxes.

For me this was a kind of mystery for some time. However, after thinking about the result of my small test-program, I solved the problem.

MDI-parent and Ctrl-C

The reason for this phenomenon is that the MDI-parent receives these key events first. As there is no code to handle these events in the way you want, the application just ignores the whole event, leaving the user in despair.

The solution is simple, in contrary to defining the reason for the problem, as usual. Place the following code in the top of the code-behind of the MDI-Parent:

```
'Use a Windows-api call to forward the triggered event  
Private Declare Auto Function SendMessage Lib "user32" _
```

```
(ByVal hwnd As IntPtr, ByVal wParam As Integer, ByVal lParam As IntPtr, ByVal IPParam As  
IntPtr) As IntPtr
```

Enable Ctrl-C, Ctrl-X and Ctrl-V in textbox with Visual Basic.Net

Written by Joris Bijvoets

Tuesday, 24 November 2009 21:15 - Last Updated Friday, 26 November 2010 19:22

```
'Override the Pressed Key Processing Routine of the MDI-Parent  
Protected Overrides Function ProcessCmdKey _
```

```
(ByRef msg As System.Windows.Forms.Message, ByVal keyData As  
System.Windows.Forms.Keys) As Boolean
```

```
SendMessage(Me.ActiveMdiChild.Handle, msg.Msg, msg.WParam, msg.LParam)
```

```
End Function
```

After this, your MDI-Childs automatically handle the **Ctrl-C, Ctrl-X and Ctrl-V** etc. as you want.

So DON'T accept the solution to write event handlers for each textbox, with code like:

```
If KeyCode = 67 And Shift = vbCtrlMask Then  
Clipboard.Clear  
Clipboard.SetText SomeTextbox.Text  
End If
```