

Forum:[Hack Forums](#)

Thread: [User assigned hotkeys](#)

MrRapidz - May 21, 2015, 12:46 pm

Okay so I've been coding a little project today.

I'm stuck at this part though, I want the user to be able to set customized hotkeys for the application.

Example:

In settings theres a textbox.

When they focus this textbox and click any key I need to detect what key is pressed and put it into the textbox so I can save the key for use.

I basically need to detect the name of the key pressed, the key's name needs to be the same name that it uses in code though (ex. 1's name in code is D1).

I hope that made sense.

How can I achieve this? Google was to no help.

RandomUser123 - May 21, 2015, 1:04 pm

(May 21, 2015 12:46 PM)MrRapidz Wrote: [\[CITE\]](#) Okay so I've been coding a little project today.

I'm stuck at this part though, I want the user to be able to set customized hotkeys for the application.

Example:

In settings theres a textbox.

When they focus this textbox and click any key I need to detect what key is pressed and put it into the textbox so I can save the key for use.

I basically need to detect the name of the key pressed, the key's name needs to be the same name that it uses in code though (ex. 1's name in code is D1).

I hope that made sense.

How can I achieve this? Google was to no help.

Im searching for the same and.. yea.. google doesn't help.

Already asked for the same in this section and got no useful replies, i hope u get a good one (which will be useful for me too ;'D)

MrRapidz - May 21, 2015, 1:13 pm

(May 21, 2015 01:04 PM)RandomUser123 Wrote: [\[CITE\]](#) Im searching for the same and.. yea.. google doesn't help.

Already asked for the same in this section and got no useful replies, i hope u get a good one (which will be useful for me too ;'D)

Yea lets hope some hero answers :D

\$uicidal - May 21, 2015, 1:23 pm

Dim Key As Keys

Key = Textbox1.Text

You'll have to get the Long value of the key instead of the string.

You can see the values by hovering over the key in the Keys. context menu.

That's just setting the key though, I'm not sure about detecting the key.

You can probably use GetAsyncKeyState or a keyboard hook would be better.

Not sure how to translate that though.

MrRapidz - May 21, 2015, 1:30 pm

(May 21, 2015 01:23 PM)Lawlly Wrote: [\[CITE\]](#) Dim Key As Keys

Key = Textbox1.Text

You'll have to get the Long value of the key instead of the string.

You can see the values by hovering over the key in the Keys. context menu.

That's just setting the key though, I'm not sure about detecting the key.

You can probably use GetAsyncKeyState or a keyboard hook would be better.

Not sure how to translate that though.

Its detecting the key that I'm lost on, also I don't need to detect the keypress globally so keyboard hook / GetAsyncKeyState wouldn't be the right way to do it. Atleast, I don't think so :s

I found out how to detect keychars though but this only includes (1-9 / a-z) as far as I can see. Also IDK how to translate it to the key's code.

\$uicidal - May 21, 2015, 1:32 pm

The best I can offer you is manually setting the key, other than that, I can't do anything

NaMeMaN - May 21, 2015, 4:23 pm

Code Private Declare Function GetAsyncKeyState Lib "user32" (ByVal vkey As Integer) As Short
Private Sub TextBox1_KeyPress(sender As Object, e As KeyPressEventArgs) Handles

TextBox1.KeyPress

Dim Chrctr As Integer

For x = 1 To 255

Chrctr = 0 : Chrctr = GetAsyncKeyState(x)

If Chrctr = -32767 Then : RichTextBox1.Text += Chr(x) & vbNewLine : End If

Next x

End Sub

i guess im a hero now :D

iGuessSo - May 21, 2015, 5:31 pm

You shouldn't need to use anything like NaMeMaN suggests from what I can tell. Just create a KeyDown event and it will detect keypresses.

Code Private Sub TextBox1_KeyDown(sender As Object, e As KeyEventArgs) Handles
TextBox1.KeyDown

TextBox1.Text = e.KeyData.ToString()

End Sub

The format isn't probably what you're expecting, but you can easily format that. ex:

Ctrl + Shift + F --> F, Shift, Control

a --> A

Backspace -> Back

Ctrl + Alt + B --> B, Control, Alt

NaMeMaN - May 21, 2015, 7:24 pm

(May 21, 2015 05:31 PM)iGuessSo Wrote: [\[CITE\]](#) You shouldn't need to use anything like NaMeMaN suggests from what I can tell.

(May 21, 2015 12:46 PM)MrRapidz Wrote: [\[CITE\]](#) I basically need to detect the name of the key pressed, the key's name needs to be the same name that it uses in code though (ex. 1's name in code is D1).

He wants to 'detect' the key pressed and have use of the key's code.

```
For x = 1 To 255
Chrctr = 0
Chrctr = GetAsyncKeyState(x)
If Chrctr = -32767 Then
RichTextBox1.Text += Chr(x) & vbNewLine
End If
Next x
```

X is the integer value of the character being detected.

A = 65, B = 66, C = 67, D = 68, etc...

From what he said in his post, I believe what I have given him is more than sufficient to satisfy the desires of the OP. My first thoughts were he was wanting to make a customizable hotkey based on the detected key, but then after reading through his post I realized he was simply asking to save the key presses so I came up with this.

Odus - May 22, 2015, 2:56 am

(May 21, 2015 04:23 PM)NaMeMaN Wrote: [\[CITE\]](#) Code If Chrctr = -32767 Then : RichTextBox1.Text += Chr(x) & vbNewLine : End If

Please don't do this. Oh, and that entire code can be simplified to:

Code e.KeyCode

Anyway, OP said entering 1 would produce "D1". AFAIK, this gives exactly that:

Code e.KeyCode.ToString()

Used in the KeyDown event.

MrRapidz - May 22, 2015, 3:19 am

(May 22, 2015 02:56 AM)Odus Wrote: [\[CITE\]](#) Please don't do this. Oh, and that entire code can be simplified to:

Code e.KeyCode

Anyway, OP said entering 1 would produce "D1". AFAIK, this gives exactly that:

Code e.KeyCode.ToString()

Used in the KeyDown event.

This worked, damn. So much effort for that xD