
Subject: Push button by keystroke (without ALT)
Posted by [zeiler](#) on Tue, 30 Nov 2010 08:49:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello,

I'm new to the forum (and the UPP-Framework). I really like both Nice work, the people are friendly, the answers helpful. I already searched the forum for my problem but didn't find the exact problem I have. Now here's my problem: I need the ability to push a button or executing the corresponding button callback when a key is pressed (without the ALT key).

My current solution is to derive a class from Upp::Button calling it MyButton and adding a function to that class which returns the protected Button-variable accesskey. Then I overwrite the main Windows Key()-function and add call button.PseudoPush when the access key is pressed.

Well, maybe I'll better show you the code:

In the header file I have the following:

```
class MyButton : public Button {
public:
    dword GetAccessKey();
};
```

```
dword MyButton::GetAccessKey()
{
    return ToLower((wchar)accesskey);
}
```

In the main Windows class declaration I add the button:

```
MyButton button;
```

In the main Windows constructor:

```
button.SetLabel("E&xit");
```

And finally in the overwritten Key()-function of the main Window:

```
bool MyMainWindow::Key(dword key, int count)
{
    if ( button.GetAccessKey() == key )
    {
        button.PseudoPush();
        return true;
    }
    return TopWindow::Key(key, count);
}
```

Now here's my question:

Is there a better, cleaner, more elegant solution to this problem?

Best regards,
Julian

Subject: Re: Push button by keystroke (without ALT)
Posted by [dolik.rce](#) on Tue, 30 Nov 2010 10:31:54 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Julian,
welcome here

You can trigger the button action from a Key() handler of your application. Have a look at its description in Ctrl reference.

A simple example would be:

```
#include <CtrlLib/CtrlLib.h>
using namespace Upp;
```

```
class Win: public TopWindow{
    typedef Win CLASSNAME;
public:
    Button b;
    Win(){
        Add(b.HCenterPos().VCenterPos());
        b.SetLabel("Close");
        b<<=THISBACK(OnPush);
    }
    void OnPush(){
        Exclamation("Button pushed, exiting...");
        Close();
    }
    bool Key(dword key,int){
        if(key=='q'){
            b.Action();
            return true;
        }
        return false;
    }
};
```

```
GUI_APP_MAIN{
    Win().Run();
}
```

Another possible solution could use InstallKeyHook() which works for entire app, not only one window. It is documented in the same document.

Best regards,
Honza

Subject: Re: Push button by keystroke (without ALT)
Posted by [koldo](#) on Tue, 30 Nov 2010 10:47:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

Or this?

```
class MyButton : public Button {
private:
virtual void LeftDown(Point p, dword key) {
    Button::LeftDown(p, key);
    if (key & K_ALT)
        PromptOK("Done");
}
};
```

Subject: Re: Push button by keystroke (without ALT)
Posted by [mirek](#) on Wed, 01 Dec 2010 16:55:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

dolik.rce wrote on Tue, 30 November 2010 05:31Hi Julian,
welcome here

You can trigger the button action from a Key() handler of your application. Have a look at its description in Ctrl reference.

There is even the nice method "PseudoPush" that makes push animation...

Subject: Re: Push button by keystroke (without ALT)
Posted by [zeiler](#) on Mon, 06 Dec 2010 08:05:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thank you all for your help and sorry for replying so late.

Julian
