

---

Subject: trouble with tabs

Posted by [mtdew3q](#) on Wed, 02 Oct 2019 00:15:10 GMT

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

---

Hi all-

I am very puzzled. i want code to activate on tab WhenSet, and then to identify which tab is the current tab.

Am I doing something wrong?

```
tab1.Add(eventGrid.SizePos(), t_("Events"));
tab1.Add(dateGrid.SizePos(), t_("Dates"));
tab1.Add(sqlArr.SizePos(), t_("Tasks"));

tab1.WhenSet = THISBACK(ManageMenus);

void CantonEdu::ManageMenus() {
// TabCtrl::Item & tabItemZero = tab1.GetItem(0);

int n3 = tab1.Get();
bool n2 = tab1.IsAt(eventGrid);

// PromptOK( "" + n3);
// PromptOK( "" + n2);
//n3 and n2 do not return anything (blank dialog msg window).
// cant use tabItemZero because I can't pass in an int like n3 to GetItem;

// NOTE this function doesn't really do anything (yet). It is a mishmash of pseudo code.
// I used examples of Get and IsAt to communicate that the PromptOk dialog window returns
nothing.

}
```

I will check back in about 14 hours. I am at GMT -5.

thnx.  
roboloki

---

---

Subject: Re: trouble with tabs

Posted by [koldo](#) on Wed, 02 Oct 2019 06:52:02 GMT

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

---

This works for me:

```
tab.Add(menuOpen.SizePos(), "Open");
tab.Add(menuConvert.SizePos(), "Convert");
```

```
tab.WhenSet = [&] {
    PromptOK("Selected tab " + FormatInt(tab.Get()));
    if (tab.IsAt(menuOpen))
        PromptOK("At Open");
    else if (tab.IsAt(menuConvert))
        PromptOK("At Convert");
};
```

You can even get direct access to the tabs, in this case, to disable it:  
TabCtrl::Item& tabOpen = tab.GetItem(tab.Find(menuOpen));  
tabOpen.Enable(false);

---

---

Subject: Re: trouble with tabs  
Posted by [mtdew3q](#) on Wed, 02 Oct 2019 11:32:17 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi -

It is a scoping issue. I will figure it out after work. I see that your event handler scope worked.

thanks for the cool tip,  
roboloki

---

---

Subject: Re: trouble with tabs  
Posted by [mtdew3q](#) on Thu, 03 Oct 2019 00:49:59 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Koldo-

I checked out function references, and I can't get them in the same scope as the GUI.

Is there an easy way with a function reference or lamda event handler to pass in a parameter that is the same scope as the GUI?

I only have scope on the lamda for the tab. People might like multiple parameters in the same scope with their controls in the layout.

I know that is asking a lot, and if you are busy I understand.

I can always use a different way to solve the problem.

I bet some other people are wondering how to use multiple parameters from a GUI in a thisback

or lamda thisback too.

thanks,  
roboloki

---

---

Subject: Re: trouble with tabs  
Posted by [koldo](#) on Thu, 03 Oct 2019 06:12:27 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello Roboloki

I am sorry, I do not understand you well. Could you explain it even simpler, with a tiny sample? You can always pass arguments to a callback (remember THISBACK1, THISBACK2, ...) or to a lambda.

---

---

Subject: Re: trouble with tabs  
Posted by [mtdew3q](#) on Thu, 03 Oct 2019 22:04:08 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Koldo-

I outlined 2 examples to illustrate my confusion.

I hope you can now see what I am doing wrong.

Example 1:

```
int func(int a) {  
    // cout << "Hello" << a << '\n';  
    return 2;  
}
```

```
struct CantonEdu : public WithRooLayout<TopWindow> {  
    WithQueryLayout<TopWindow> query;
```

```
class functions..  
class members...
```

```
void CantonEdu::ManageMenus(int n);  
Button b1;  
TabCtrl tab1 // defined in layout  
int func(int a);
```

```
typedef CantonEdu CLASSNAME;  
};
```

```
int (& r_ref)(int) = func;  
Add ( b1.LeftPos ( 10, 100 ).BottomPos ( 10, 30 ) );  
b1.SetLabel ( "Click me!" );  
b1 <<= THISBACK ( r_ref, 1 ) ; /** thisback and r_ref not agreeing in this scope (GUI) **/
```

b1 and r\_ref not in same scope.

Example 2:

```
tab1.WhenSet = [&] {  
  
    int n = tab1.Get(); // compiles  
    ManageMenus(n); // compiles  
  
};
```

```
void CantonEdu::ManageMenus(int n) {  
    PromptOK( " " + n): // (blank PromptOk - empty value)  
}
```

n does not equal index of recently clicked tab

I'd like to be able thisback to call function with these parameters as well: void ManageMenus (TabCTRL tab1, MenuBar menu).

```
tab1.WhenSet = THISBACK(ManageMenus,tab1, menu);
```

NOTE: I mean to indicate a function reference that points to function like:

```
void func (CtrlTab tab, MenuBar menu) ;
```

```
void (& func_ref)(CtrlTab, MenuBar) = func;  
tab1.WhenSet = THISBACK(func_ref, tab1, menu);
```

I hope that is a little more clear.

I don't know if these examples are easy to follow and/or easy to implement.

Thnx for any assistance.

roboloki

---

Subject: Re: trouble with tabs  
Posted by [mtdew3q](#) on Sun, 06 Oct 2019 16:06:34 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Koldo-

I got knocked out of U++ this past couple of days by busy moments at work and elsewhere.

I had some time today to look at my code and got it working.

I wanted a system where the menus changed based on which tab is selected.

Here is the code for that.

Thanks so much for replying last time, and sorry for being confused.

Below is the code that is now working. THNX !

```
#include <CtrlLib/CtrlLib.h>
#include "cookie.h"

void Blam::Setup ()
{

    CtrlLayout ( *this, String ( "Blam" ) + " 1.0" );

    tab.Add ( a1.SizePos(), "a1" );
    tab.Add ( a2.SizePos(), "a2" );

    AddFrame ( menu ).LeftPosZ ( 0, 304 ).TopPosZ ( 0, 15 );

    menu.Set ( THISBACK ( MainMenu1 ) );

    tab.WhenSet = [&]
    {
        PromptOK ( "Selected tab " + FormatInt ( tab.Get() ) );

        if ( tab.IsAt ( a1 ) )
```

```

{
    menu1.Clear();
    PromptOK ( "At a1" );
    menu.Set ( THISBACK ( MainMenu1 ) );

}

else
    if ( tab.IsAt ( a2 ) )
    {
        PromptOK ( "At a2" );
        menu.Clear();
        menu.Set ( THISBACK ( MainMenu2 ) );

    }

};

}

void Blam::MainMenu1 ( Bar& bar )
{
    bar.Sub ( "cookie 1", [=] ( Bar & bar )
    {
        bar.Add ( "eat cookie 1", THISBACK ( func1 ) );

    }

    );
}

void Blam::MainMenu2 ( Bar& bar )
{
    bar.Sub ( "cookie 2", [=] ( Bar & bar )
    {
        bar.Add ( "eat cookie 2", THISBACK ( func2 ) );

    }

    );
}

```

```
}
```

```
GUI_APP_MAIN
```

```
{  
  Blam b;  
  b.Setup();  
  b.Run();
```

```
}
```

---

Subject: Re: trouble with tabs

Posted by [koldo](#) on Mon, 07 Oct 2019 06:31:21 GMT

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

---

Hello Roboloki

Yes, I would do it your way.

In addition, it is possible to join the menu functions in one. For example:

```
tab.WhenSet = [&] {  
  PromptOK("Selected tab " + FormatInt(tab.Get()));
```

```
  if (tab.IsAt(a1)) {  
    PromptOK("At a1");  
    menu.Set(THISBACK1(MainMenu, 1));  
  } else if (tab.IsAt(a2)) {  
    PromptOK("At a2");  
    menu.Set(THISBACK1(MainMenu, 2));  
  }  
};
```

...

```
void Blam::MainMenu(Bar& bar, int num) {  
  bar.Sub(Format(t_("Cookie %d"), num), [=](Bar & bar, int num) {  
    bar.Add(Format(t_("Eat cookie %d"), num), THISBACK1(func, num));  
  });  
}
```