
Subject: Using value from EditInt
Posted by [BenP](#) on Sun, 30 Apr 2006 19:15:06 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

I am creating a helpdesk with U++ and I have a dialog box where I will enter a ticket ID to lookup(below)

I then have a function EditTicket(int ticket_id) which retrieves the ticket info and allows me to edit it but I have a problem with passing what is in the edit box to my function.

I have tried the following but with no success

```
d.TicketLookupBtn <<= THISBACK1(EditTicket, d.TicketLookupBox);
```

How do I use what is in the edit box and pass it to my function?

Subject: Re: Using value from EditInt
Posted by [mirek](#) on Sun, 30 Apr 2006 20:27:29 GMT
[View Forum Message](#) <> [Reply to Message](#)

```
void MyApp::EditTicket()  
{  
    int q = ~ticketLookupBox;  
}
```

```
MyApp::MyApp()  
{  
    TicketLookupButton <<= THISBACK(EditTicket);  
}
```

Hope that helps..

Mirek

Subject: Re: Using value from EditInt
Posted by [forlano](#) on Sun, 30 Apr 2006 21:52:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

BenP wrote on Sun, 30 April 2006 21:15Hi,
I have tried the following but with no success
d.TicketLookupBtn <<= THISBACK1(EditTicket, d.TicketLookupBox);

How do I use what is in the edit box and pass it to my function?

To get the data associated with the widget myWidget you can use the operator '~'. For example a = ~myWidget as showed above.

When you need to set some value use the operator '<<='. For example:

```
myWidget <<= data;
```

In your case I think you can use

```
d.TicketLookupBtn <<= THISBACK1(EditTicket, ~d.TicketLookupBox);
```

Let us know if you resolved the problem.

Luigi

Subject: Re: Using value from EditInt

Posted by [mirek](#) on Sun, 30 Apr 2006 22:05:29 GMT

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

forlano wrote on Sun, 30 April 2006 17:52BenP wrote on Sun, 30 April 2006 21:15Hi,

I have tried the following but with no success

```
d.TicketLookupBtn <<= THISBACK1(EditTicket, d.TicketLookupBox);
```

How do I use what is in the edit box and pass it to my function?

To get the data associated with the widget myWidget you can use the operator '~'. For example a = ~myWidget as showed above.

When you need to set some value use the operator '<<='. For example:

```
myWidget <<= data;
```

In your case I think you can use

```
d.TicketLookupBtn <<= THISBACK1(EditTicket, ~d.TicketLookupBox);
```

Let us know if you resolved the problem.

Luigi

No, this is wrong - you would call EditTicket with TicketLookupBox value extracted at the moment of THISBACK1 creation, not the current value!

Mirek

Subject: Re: Using value from EditInt

Posted by [BenP](#) on Mon, 01 May 2006 10:19:29 GMT

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

Neither of the above examples worked I will give snippets of my code in the hope it may help. LookupTicket() is the function which shows the lookup dialog(see first post) and EditTicket(int ticket_id) will give me a dialog to update the ticket. The ticket_id parameter must be passed to the EditTicket function so it can get data for the correct ticket.

```

void Helpdesk::LookupTicket()
{
    WithLookupTicket<TopWindow> d;
    CtrlLayout(d, "Lookup Ticket");

    //This is the problem code
    d.TicketLookupBtn <<= THISBACK1(EditTicket, d.TicketLookupBox);
    //=====

    d.Run();
}

void Helpdesk::EditTicket(int ticket_id)
{
    WithEditTicket<TopWindow> d;
    CtrlLayout(d, "Edit Ticket");

    //Edit Ticket Code

    d.Run();
}

```

Subject: Re: Using value from EditInt
 Posted by [mirek](#) on Mon, 01 May 2006 11:55:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

BenP wrote on Mon, 01 May 2006 06:19 Neither of the above examples worked I will give snippets of my code in the hope it may help.
 LookupTicket() is the function which shows the lookup dialog(see first post) and EditTicket(int ticket_id) will give me a dialog to update the ticket. The ticket_id parameter must be passed to the EditTicket function so it can get data for the correct ticket.

```

void Helpdesk::LookupTicket()
{
    WithLookupTicket<TopWindow> d;
    CtrlLayout(d, "Lookup Ticket");

    //This is the problem code
    d.TicketLookupBtn <<= THISBACK1(EditTicket, d.TicketLookupBox);
    //=====

    d.Run();
}

void Helpdesk::EditTicket(int ticket_id)
{

```

```
WithEditTicket<TopWindow> d;  
CtrlLayout(d, "Edit Ticket");
```

```
//Edit Ticket Code
```

```
d.Run();  
}
```

Ah, I see. This is not typical solution, the problem is that still that "THISBACK1" used this way simply stores current value for EditTicket.

There are several ways how to solve this:

- use class for LookupTicket, then the EditTicker, parameterless, will be its member and you will be able to read the value of LookupTicket's LookupTicketBox as it will be the member. This is what I proposed in recent reply.

- use breaker - if you press that button, "d.Run" will be broken and you will be able to read the value of d.TicketLookupBox and call EditTicket.

```
d.Breaker(d.TicketLookupBtn, 111);  
for(;;) {  
    int c = d.Run();  
    if(c == 111)  
        EditTicket(~d.TicketLookupBox);  
    if(c == IDCANCEL) // close button will send this  
        break;  
}
```

- you could also make EditTicket to accept a pointer to EditInt and pass that pointer using THISBACK1. This is the worst solution IMHO.

Mirek
