
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 EditTicket, 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 breaked 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
