
Subject: OK and Cancel Layout

Posted by [sduensin](#) on Wed, 21 Aug 2024 23:55:07 GMT

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

I've just started using U++ and so far am impressed. However, I've run into some confusion with laying out dialogs that contain "OK" and "Cancel" buttons.

The tutorial says to name two buttons - one "ok" and one "cancel" - and then call CtrlLayoutOKCancel. I did this, and it works. But...

Can you not control the order of the buttons? On my form, I have "cancel" and then "ok". When it displays at runtime, the "ok" button is first! I can deal with this, except when using PromptOKCancel, they're the other way around!

How can I have consistent buttons?

Subject: Re: OK and Cancel Layout

Posted by [koldo](#) on Fri, 23 Aug 2024 05:43:29 GMT

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

Hi sduensin

Please upload your sample code and we will show you how easy is doing what you want.

Subject: Re: OK and Cancel Layout

Posted by [sduensin](#) on Fri, 23 Aug 2024 21:33:10 GMT

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

Thanks! I've attached my current code. When you run it, if you pull down "Settings" and go to the database setup, you'll see the buttons are the reverse from how they are displayed if you pull down "File" and select exit.

Also, is there a way to change dialog titles? Right now I'm calling the app "Store Manager" but PromptOK() and friends display "StoreManager" (no space) in the title bar.

I appreciate the help!

File Attachments

1) [StoreManager.tar.gz](#) , downloaded 107 times

Subject: Re: OK and Cancel Layout

Posted by [koldo](#) on Sun, 25 Aug 2024 18:26:19 GMT

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

Hi sduensin

For some reason there are two application names, one defined in App.cpp (static char sAppName[256]) and other in Ctrl.cpp (String Ctrl::appname). To be sure, I will use both, so I would leave main like this:

```
GUI_APP_MAIN {  
    StoreManager app;  
    app.SetAppName(APP_NAME);  
    SetAppName(APP_NAME);  
    app.Run();  
}
```

In addition I cannot see the problem with the buttons; they appear as they are designed. Look at this picture:

File Attachments

1) [theide_AefZwetjsn.png](#), downloaded 351 times

Subject: Re: OK and Cancel Layout

Posted by [sduensin](#) on Mon, 26 Aug 2024 01:00:21 GMT

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

Must be a Linux/GTK issue...

File Attachments

1) [Screenshot from 2024-08-25 19-57-05.png](#), downloaded 345 times

Subject: Re: OK and Cancel Layout

Posted by [koldo](#) on Mon, 26 Aug 2024 07:34:54 GMT

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

Oh yes, probably that is enforced by GTK.

I advise you to simply replace the CtrlLayoutOKCancel() by a CtrlLayout(), and handling the OK and Cancel buttons.

File Attachments

1) [img_msrdc_pGQKCCGT5J.png](#), downloaded 340 times

Subject: Re: OK and Cancel Layout

Posted by [sduensin](#) on Mon, 26 Aug 2024 22:16:19 GMT

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

[quote title=koldo wrote on Sun, 25 August 2024 13:26]

For some reason there are two application names, one defined in App.cpp (static char sAppName[256]) and other in Ctrl.cpp (String Ctrl::appname). To be sure, I will use both, so I would leave main like this:

```
GUI_APP_MAIN {  
    StoreManager app;  
    app.SetAppName(APP_NAME);  
    SetAppName(APP_NAME);  
    app.Run();  
}
```

That fixed it! Thanks!

Subject: Re: OK and Cancel Layout

Posted by [koldo](#) on Tue, 27 Aug 2024 06:15:46 GMT

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

You are welcome!

Subject: Re: OK and Cancel Layout

Posted by [sduensin](#) on Wed, 28 Aug 2024 22:31:14 GMT

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

Doh! I just noticed that this code, while it does fix the config dir name and app name, it does not fix where the log is written. Oh well.

Subject: Re: OK and Cancel Layout

Posted by [mirek](#) on Sun, 01 Sep 2024 22:24:45 GMT

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

sduensin wrote on Mon, 26 August 2024 03:00Must be a Linux/GTK issue...

There is very special and sort of nasty code that swaps OK and Cancel buttons automatically in gtk and macos because that is how those platform have these buttons usually. I mean, you desing as "in windows" and it then gets swapped in gtk.

If you really disagree, you can disable this behaviour with SwapOKCancel_Write(false).
