

---

Subject: Pointer on Labels defined in lay.h  
Posted by [Yeti86](#) on Sat, 22 Feb 2014 11:43:33 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello Everyboy!

Is there a possibility to point on a Label or a Button which are created through the layout?  
I get an error message(okay it's in german ) :

error C2248: "Upp::Ctrl::operator =": Kein Zugriff auf private Member, dessen Deklaration in der Upp::Ctrl-Klasse erfolgte.  
->something like no acces to private member, which are declared in Upp::Ctrl-class

The reason why I want to point on the Labels and button is that in my program every button does something which should be printed in the label. It's the same function called from the buttons but for differennt labels. So I want to call the function with pointers to the different labels and buttons.

Greetings

Yeti

---

---

Subject: Re: Pointer on Labels defined in lay.h  
Posted by [Oblivion](#) on Sat, 22 Feb 2014 14:34:19 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello Yeti,

If I understand you correctly, you want to manipulate the ctrls created within layout editor, right?

This is quite easy. When you create, say, a label or a button, or whatever ctrl you like in layout editor, first you should give it a name in the layout editor. Then use it like ordinary ctrls in your code.

For example in the below screenshot I named it "mylabel". When I want to change its text, I simply call "mylabel.SetText("foo");" in code. That's all.

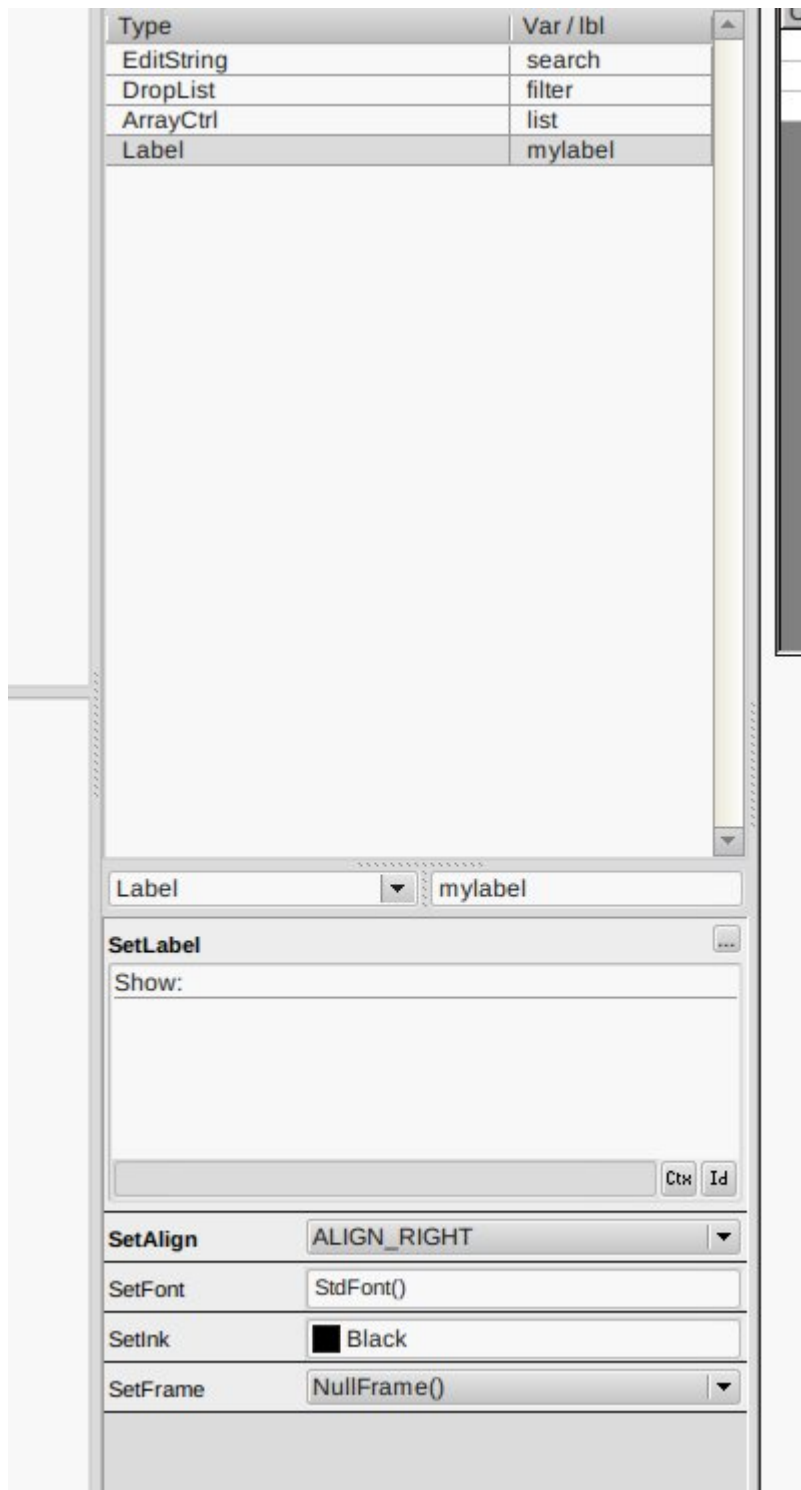
I suggest you reading the examples and reference code in UPP source. They are usually self explanatory and will save your time a lot.

Regards.

## File Attachments

---

1) [screenshot1.jpg](#), downloaded 603 times



---

Subject: Re: Pointer on Labels defined in lay.h  
Posted by [Yeti86](#) on Sat, 22 Feb 2014 19:33:14 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello Oblivion!

Thanks for the reply, but that's not the problem.

Attached you find my layout file. So I have a lot of Buttons and labels which should do all the same, for one instance of my playerclass. For less coding effort I tried to define pointer arrays on the different labels and buttons.

```
#include "Scoutfunctions.hpp"
using namespace Upp;

#define LAYOUTFILE <blala/blala.lay>
#include <CtrlCore/lay.h>
class blala : public WithblalaLayout<TopWindow> {
public:
    typedef blala CLASSNAME;
    blala();
    ~blala();
    // Label *p_playernames;
    //Label *p_playernumbers;
    Button *p_2points;
    Button *p_3points;
    Button *p_zero_of_one;
    Button *p_one_of_one;
    Label *p_score_value;
    Label *p_freethrows_value;
    Label *p_threesmade;
    playerspace::player* p_Player;
    void set_attributes();
    void push_button();
};
```

```
blala::blala()
{
    p_playernames=new Label[12];
    p_playernumbers=new Label[12];
    p_2points=new Button[12];
    p_3points=new Button[12];
    p_zero_of_one=new Button[12];
    p_one_of_one=new Button[12];
    p_score_value=new Label[12];
    p_freethrows_value=new Label[12];
    p_threesmade=new Label[12];
    p_Player=new playerspace::player[12];
    p_playernames[0]=Player1;
    p_playernames[1]=Player2;
    p_playernames[2]=Player3;
    p_playernames[3]=Player4;
```

```
p_playernames[4]=Player5;  
p_playernames[5]=Player6;  
p_playernames[6]=Player7;  
p_playernames[7]=Player8;  
p_playernames[8]=Player9;  
p_playernames[9]=Player10;  
p_playernames[10]=Player11;  
p_playernames[11]=Player12;
```

Or is there even a smarter solution, like create a new own class with one row of necessaty buttons and labels?

And than define the different columns with this new class? Would it be possible?

I hope the problem is now explained

Regards

## File Attachments

1) [GUIlayout.png](#), downloaded 513 times

|            |  |            |  |  |  |  |  |  |  |
|------------|--|------------|--|--|--|--|--|--|--|
| Playtime + |  | Playtime - |  |  |  |  |  |  |  |
| Playtime:  |  | Playtime   |  |  |  |  |  |  |  |

|         |         |          |          |       |     | Score | Freethrows | 3-Points made |
|---------|---------|----------|----------|-------|-----|-------|------------|---------------|
| Player1 | Player1 | 2 Points | 3 Points | 0 / 1 | 1/1 | Score | Freethrows | 3-Points made |
| Player1 | Player1 | 2 Points | 3 Points | 0 / 1 | 1/1 | Score | Freethrows | 3-Points made |
| Player1 | Player1 | 2 Points | 3 Points | 0 / 1 | 1/1 | Score | Freethrows | 3-Points made |
| Player1 | Player1 | 2 Points | 3 Points | 0 / 1 | 1/1 | Score | Freethrows | 3-Points made |
| Player1 | Player1 | 2 Points | 3 Points | 0 / 1 | 1/1 | Score | Freethrows | 3-Points made |
| Player1 | Player1 | 2 Points | 3 Points | 0 / 1 | 1/1 | Score | Freethrows | 3-Points made |
| Player1 | Player1 | 2 Points | 3 Points | 0 / 1 | 1/1 | Score | Freethrows | 3-Points made |
| Player1 | Player1 | 2 Points | 3 Points | 0 / 1 | 1/1 | Score | Freethrows | 3-Points made |
| Player1 | Player1 | 2 Points | 3 Points | 0 / 1 | 1/1 | Score | Freethrows | 3-Points made |
| Player1 | Player1 | 2 Points | 3 Points | 0 / 1 | 1/1 | Score | Freethrows | 3-Points made |
| Player1 | Player1 | 2 Points | 3 Points | 0 / 1 | 1/1 | Score | Freethrows | 3-Points made |

---

Subject: Re: Pointer on Labels defined in lay.h  
 Posted by [Oblivion](#) on Sat, 22 Feb 2014 22:01:33 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello Yeti,

First of all, never allocate any ctrl object (or, for that matter any object) from the heap. Just avoid new and delete unless it is necessary. Use pointers only to point objects, not for memory management. Get yourself familiar with the core types and NTL containers, such as arrays and vectors, they are your best friend (and they will handle memory de/allocation, most of the time). Just read the docs, both api and other. They are straightforward and easy to understand. Once you use the ntl containers, you will never want to do otherwise.

There are several ways you could achieve what you ask. But the easiest possible solution that comes to my mind now is using a ParentCtrl to represent a row (as far as I understand, a row in your code is a visual representation of a players' variables). So my suggestion would be:

- 1) Create a class that represents a row (name it whatever you like, a player, etc). It is pretty straightforward.
- 2) Add a layout to it (and add the children (buttons, labels, etc.) of the parent ctrl (a "Row") in the layout editor).
- 3) Since you now have a row class, in your main windows layout, add Row as custom ctrls. In layout editor -> Press right mouse button -> add a "Custom Class" and then set the custom class name (Row) and its ID (row1, row2, etc...)
- 4) done.

This way, your code will be much less cluttered and more logical, and readable. I attached the source code of an example that contains three rows (each with a label and two buttons). The example code does nothing fancy but I hope it will give you the idea, so please take your time and examine it.

Ps. A better approach, if you'd have a lot of rows, say, 20, would be to use Arrays (eg., `Array<Row> ...` ). But it's another story.

Regards.

## File Attachments

1) [ParentCtrlExample.zip](#), downloaded 348 times

---

---

Subject: Re: Pointer on Labels defined in lay.h  
Posted by [Yeti86](#) on Sun, 23 Feb 2014 13:39:59 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello Oblivion,

thank you very much for this great example and explanation!!!

I really appreciate it!

---