Quote:
I felt that... but I haven't been successful. What would be the correct way then?

```cpp
#include <CtrlLib/CtrlLib.h>

class MyClassWindow : public TopWindow {
private:
 StatusBar  status1; //edited: was MyStatusBar
      InfoCtrl  info2;

public:
 typedef MyClassWindow CLASSNAME;
 MyClassWindow();
};


MyClassWindow::MyClassWindow()
{
 status1.AddFrame(info2.Width(250));
 info2="info2: Welcome to the Ultimate++ !";
 AddFrame(status1.Height(25));

}


GUI_APP_MAIN
{
  MyClassWindow().Title("MyClassWindow1").Zoomable().Sizeable().Run();
  MyClassWindow().SetRect(0, 0, 260, 80);
}
```

Note once again that this is basically the wrong code - I am just demonstrating that in similar situation, creating separate class is not necessarry - and not creating it is better, as you will have to access that "info2" from MyClassWindow anyway.

The lesson to learn is that in U++, class composition does not have to follow GUI hierarchy. It is not uncommon to e.g. have TopWindow as member variable and put "self" into it:

```cpp
struct Foo : Button {
  TopWindow win;
```

```
  void Perform() { win.Run(); }

  Foo() {
    win.Add(SizePos());
  }
};
```