
Subject: Re: How do I create a loop for a window to react to volatile changes to a global variable?

Posted by [slashuip](#) on Mon, 11 Mar 2019 12:29:26 GMT

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

Hi Oblivion

What I do is more like this:

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

```
using namespace Upp;
```

```
class ThreadTest;
```

```
void RunThread(ThreadTest *pTT);
```

```
class ThreadTest : public TopWindow
```

```
{
```

```
    Button  start;
```

```
    ArrayCtrl array;
```

```
public:
```

```
    ThreadTest()
```

```
{
```

```
    SetRect(0, 0, 800, 600);
```

```
    CenterScreen();
```

```
    array.AddColumn("Number");
```

```
    Add(array.HSizePos().VSizePos(0, 30));
```

```
    Add(start.SetLabel("Start Thread").RightPos(2, 120).BottomPos(2, 26));
```

```
    start << [=] { RunThread(this); };
```

```
}
```

```
void SetArray(int n)
```

```
{
```

```
    GuiLock __;
```

```
    array.Add(n);
```

```
    if(n == 10) { // Resize window from within thread...
```

```
        SetRect(0, 0, 640, 480);
```

```
}
```

```
}
```

```
};
```

```
void RunThread(ThreadTest *pTT)
```

```
{
```

```
    Thread t;
```

```
    ThreadTest *ptt=pTT;
```

```
    t.Run([=]
```

```
{
```

```
    for(int i = 1; i < 10; i++)
```

```
{
```

```
        // GuiLock ____;  
        ptt->SetArray(i);  
    }  
});  
t.Detach();  
}
```

```
GUI_APP_MAIN  
{  
    ThreadTest().Run();  
}
```

Where the thread-func is global. This still works, and the only real difference between my app and this is that I use `std::thread` ... maybe I shouldn't
I'll change my code to use `Upp::Thread`
(and if you don't hear from me again, it worked)

thx
