
Subject: Re: While loop, sleep
Posted by [Klugier](#) on Fri, 27 Oct 2023 19:50:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello,

I agree with zsolt. SetTimeCallback is the way to go in your case. Currently, I am working on expanding our GUI tutorial about exemplary usage of that feature.

Here is the code:

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

using namespace Upp;

struct MyAppWindow : TopWindow {
    Rect rect;
    Color color;

    MyAppWindow() {
        Zoomable().SetRect(0, 0, 200, 200);

        RandomizeRect();
        SetTimeCallback(-2000, [=] { OnTimer(); });
    }

    ~MyAppWindow() {
        KillTimeCallback();
    }

    void Paint(Draw& w) override {
        Size sz = GetSize();

        w.DrawRect(sz, White());
        w.DrawRect(rect, color);
    }

    void OnTimer() {
        RandomizeRect();
        Refresh();
    }

    void RandomizeRect() {
        Size sz = GetSize();

        int length = 50;
        int x = Random() % (sz.cx - length);
        int y = Random() % (sz.cy - length);
```

```
rect = Rect(x, y, x + length, y + length);  
color = Color(Random() % 255, Random() % 255, Random() % 255);  
}  
};
```

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

The first parameter in this case mean interval in milliseconds. If it is less than zero it means that it will be repeated infinite number of times. If the number is positive it will be executed only once.
`SetTimeCallback(-2000, [=] { OnTimer(); });`

In above example, the method `OnTimer()` in `Ctrl` will be executed once per 2 seconds until cancellation. Cancellation can be done by calling `KillTimeCallback()`.

Moreover with `SetTimeCallback` you shouldn't observe any UI freezes or something like that.

Regards,
Klugier
