
Subject: ArrayCtrl.Clear() but?

Posted by [jimlef](#) on Sun, 26 Dec 2021 18:52:14 GMT

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

Merry Christmas :)

I have a simple program:

main.cpp:

```
#include "ClearTest.h"
```

```
arrayTest::arrayTest()
```

```
{  
    CtrlLayout(*this, "Window title");  
    ztest temp;  
    cancel << [=] { Close(); };  
  
    arrTest.AddColumn("Int");  
    arrTest.AddColumn("Int");  
    arrTest.AddColumn("Bool");  
  
}
```

```
void arrayTest::LoadArray( Array<ztest> t)
```

```
{  
    arrTest.Clear();  
    for (int i = 0; i < t.GetCount(); i++) if (t[i].b) arrTest.Add(t[i].t[0], t[i].t[1], (int)t[i].b);  
}
```

```
clearTest::clearTest() // using upp 15947 compiled on Linux Mint 20.2
```

```
{  
    CtrlLayout(*this, "Window title");  
    cancel << [=] { Close(); };  
    btnRun << [=] {  
        for (int i = 0; i < 10; i++) {  
            ztest *temp = new ztest;  
            srand (time(NULL));  
            temp->t[0] = i + 1;  
            temp->t[1] = i + (rand() % 100);  
            temp->b = true;  
            t.Add(pick(temp));  
        }  
        if (!aTest.IsOpen())  
            aTest.LoadArray(clone(t));  
        aTest.Open(this);  
    };  
}
```

```

GUI_APP_MAIN
{
    clearTest().Run();
}

```

ClearTest.h:

```

#ifndef _ClearTest_ClearTest_h
#define _ClearTest_ClearTest_h

```

```

#include <CtrlLib/CtrlLib.h>

```

```

using namespace Upp;

```

```

#define LAYOUTFILE <ClearTest/ClearTest.lay>
#include <CtrlCore/lay.h>

```

```

struct ztest {
    int t[2];
    bool b;
};
static Array<ztest> t;

```

```

class arrayTest : public WithMyArray<TopWindow> {
public:
    void LoadArray(Array<ztest> t);
    arrayTest();
};
class clearTest : public WithclearTestLayout<TopWindow> {
    arrayTest aTest;
public:
    clearTest();
};

```

```

#endif

```

```

ClearTest.Lay
LAYOUT(clearTestLayout, 240, 148)
    ITEM(Upp::Button, btnRun, SetLabel(t_("RunbtnRun")).LeftPosZ(28, 56).TopPosZ(76, 15))
    ITEM(Upp::Button, cancel, SetLabel(t_("Close")).LeftPosZ(148, 56).TopPosZ(76, 15))
END_LAYOUT

```

```

LAYOUT(MyArray, 400, 200)
    ITEM(Upp::ArrayCtrl, arrTest, LeftPosZ(12, 380).TopPosZ(8, 140))
    ITEM(Upp::Button, cancel, SetLabel(t_("Close")).LeftPosZ(172, 56).TopPosZ(168, 15))
END_LAYOUT

```

It compiles and runs ok - but when I click on "RunbtnRun" repeatedly, instead of the array 'Clear()'ing, it adds to bottom of list growing each time.

Am I doing something wrong here?
