
Subject: ArrayCtrl::Column lambda cleanup

Posted by [mirek](#) on Sat, 07 Oct 2017 09:59:26 GMT

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

I have noticed some lambda related overloading in ArrayCtrl::Column, so cleaned that up by adding some new methods (non-overloading). In the process, I have added reference example how to use them:

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

```
using namespace Upp;
```

```
GUI_APP_MAIN
```

```
{
    ArrayCtrl list;
    list.AddColumn("Trivial").Ctrls<EditString>(); // specify type of widget as template argument
    list.AddColumn("Factory").With( // class factory creates widget - allows customization of widget
        [](One<Ctrl>& x) {
            x.Create<Option>().NoWantFocus();
        }
    );
    list.AddColumn("Lined").WithLined( // passes the row index as the first argument
        [](int line, One<Ctrl>& x)
        {
            if(line & 1)
                x.Create<EditString>();
            else
                x.Create<Option>().SetLabel("Line " + AsString(line));
        }
    );

    for(int i = 0; i < 300; i++)
        list.Add(AsString(i), i & 1, AsString(i));
    for(int i = 1; i < 300; i += 3)
        list.GetCtrl(i, 0)->Disable();

    list.SetLineCy(Draw::GetStdFontCy() + DPI(8));

    TopWindow app;
    app.Add(list.SizePos());
    app.Sizeable();
    app.Run();
}
```

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

```
using namespace Upp;
```

```
GUI_APP_MAIN
```

```
{  
    ArrayCtrl list;
```

```
list.AddColumn("Trivial").Sorting(); // with standard comparison
```

```
list.AddColumn("With sorting lambda").SortingBy( // sort with sorting function
```

```
    [](int a, int b) -> int {  
        int q = SgnCompare(a % 100, b % 100);  
        if(q) return q;  
        return SgnCompare(a, b);  
    }
```

```
);SortDefault(); // set his column to be the initial sorting column;
```

```
list.AddColumn("Line comparison").SortingLined( // row indices passed to predicate
```

```
    [&list](int i, int j) -> bool { // sort by sum of first two columns (as an example)  
        int a = int(list.Get(i, 0)) + int(list.Get(i, 1));  
        int b = int(list.Get(j, 0)) + int(list.Get(j, 1));  
        return list.IsSortDescending() ? b < a : a < b;  
    }
```

```
);
```

```
for(int i = 0; i < 300; i++)
```

```
    list.Add((int)Random(1000), (int)Random(10000), 0);
```

```
TopWindow app;
```

```
app.Add(list.SizePos());
```

```
app.Sizeable();
```

```
list.DoColumnSort(); // sort by current sorting column (which is "With function" column because of  
SortDefault
```

```
app.Run();
```

```
}
```