
Subject: Re: Change row appearance based on column value in that row

Posted by [mirek](#) on Sun, 25 Sep 2016 21:52:53 GMT

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

slashupp wrote on Tue, 16 August 2016 12:26

- multiple indexes to same column - wtf? this throws me completely
array.AddColumn(ID2, "combined").Add(ID1).AddIndex(ID3).SetConvert(Single<MyConvert>());
ID1, ID2, ID3 ALL added to this single column - what am i missing?

Well, perhaps it is usefull to state that 'basic' operations are in fact

ArrayCtrl::AddIndex - adds 'index' (or 'column in datagrid')

ArrayCtrl::AddColumnAt - adds visual column which (at first) contains some index

ArrayCtrl::Column::Add - adds another index to the column - in this case, the value of column is ValueMap combining valus of all indices

now other methods are derived. For example

```
array.AddIndex(ID1);  
array.AddColumn(ID2, "combined").Add(ID1).AddIndex(ID3);
```

can be written using basic ops as

```
array.AddIndex(ID1);  
array.AddIndex(ID2);  
array.AddIndex(ID3);  
array.AddColumnAt(ID2, "combined").Add(ID1).Add(ID3);
```

Quote:

in MyConvert you use 0, 1 and 2 as indexes into the ValueArray - what then is the purpose of the named indexes?

Well, ArrayCtrl always creates ValueMap, but in U++, maps are ordered and ValueMap can be converted to ValueArray, that is why it is then possible to use just indexes. But you can convert to ValueMap instead and use IDs or in fact, you can use both IDs or indexes directly for Value (this was not quite possible at the time the example was written).

Hope this helps.

Mirek
