Subject: ColumnList - Set(Value key, Value val) - Working right?
Posted by jeremy_c on Sat, 21 Aug 2010 14:55:12 GMT
View Forum Message <> Reply to Message

I have the following code:

```
Value id = codelist.GetData();
codelist.Set(id, "Hello World!")
```

It does not change the selected item. Instead it seems to be changing the item at an index of "id".

This code works as expected:

```
int idx = codelist.Find(id);
codelist.Set(idx, ~d.value);
```

From Assist++ it seems that .Set(Value key, Value val) should set it based on key, thus the first example should have worked?

Jeremy

Subject: Re: ColumnList - Set(Value key, Value val) - Working right?
Posted by bonami on Fri, 08 Oct 2010 07:27:08 GMT
View Forum Message <> Reply to Message

columnlist is like a map. a key is the data's identifier not its index.
use SetIndex(index) to change current item displayed.

Subject: Re: ColumnList - Set(Value key, Value val) - Working right?
Posted by mrjt on Fri, 08 Oct 2010 09:16:09 GMT
View Forum Message <> Reply to Message

It's a very small bug. See the definition of the Set functions:

```
    void        Set(int ii, const Value& key, const Value& val, bool canselect = true);
    void        Set(int ii, const Value& key, const Value& val, const Display& display, bool canselect
= true);
    void        Set(int ii, const Value& val, bool canselect = true);
    void        Set(int ii, const Value& val, const Display& display, bool canselect = true);

    void        Set(const Value& key, const Value& val, const Display& display, bool canselect);
```

void Set(const Value& key, const Value& val, bool canselect);
You are intending to call Set(Value, Value) but there is no candiate for that, the closest being Set(Value, Value, bool). As there is an implicit declared from Value to int the compiler is actually choosing Set(int, Value), which sets the value using an index.

The quick solution is to add the canselect parameter to your call:
Value id = codelist.GetData();
codelist.Set(id, "Hello World!", true);

but the function definition should also be changed to have a default value.
