Subject: Re: sqlarray and sqlite
Posted by jimlef on Wed, 23 Sep 2020 17:57:01 GMT
View Forum Message <> Reply to Message

I realize that I've been going about this all wrong...
I've been trying to do a 1 - 1 conversion from my c# code to the c++ and U++... That's wrong
thinking.

I need to embrace the U++, so I went back to the examples, and am using the book borrowing
app. I've come up with the following code:

```
#include "customers.h"
#define MODEL <Invoices/Tables/Invoices.sch>
#include <Sql/sch_source.h>

#undef MODEL
class AddCustomer : public WithCustomerAddLayout<TopWindow> {
 public:
  SqlCtrls ctrls;
  typedef AddCustomer CLASSNAME;
  AddCustomer();
};

/*
 ITEM(Option, chkTaxable, SetLabel(t_("Taxable?")).SetFont(SansSerifZ(16)).LeftPosZ(228,
108).TopPosZ(36, 16))
 ITEM(EditString, txtCustSearch, SetFont(SansSerifZ(16)).SetFrame(FieldFrame()).LeftPosZ(136,
256).TopPosZ(404, 19))
 ITEM(EditString, txtCustName, SetFont(SansSerifZ(16)).SetFrame(FieldFrame()).LeftPosZ(136,
256).TopPosZ(76, 19))
 ITEM(EditString, txtCustEmail, SetFont(SansSerifZ(16)).SetFrame(FieldFrame()).LeftPosZ(136,
256).TopPosZ(116, 19))
 ITEM(EditString, txtCustPhone, SetFont(SansSerifZ(16)).SetFrame(FieldFrame()).LeftPosZ(136,
256).TopPosZ(156, 19))
 ITEM(EditString, txtCustAddress,
SetFont(SansSerifZ(16)).SetFrame(FieldFrame()).LeftPosZ(136, 256).TopPosZ(196, 19))
 ITEM(EditString, txtCustCity, SetFont(SansSerifZ(16)).SetFrame(FieldFrame()).LeftPosZ(136,
256).TopPosZ(232, 19))
 ITEM(EditString, txtCustState, SetFont(SansSerifZ(16)).SetFrame(FieldFrame()).LeftPosZ(136,
256).TopPosZ(272, 19))
 ITEM(EditString, txtCustZip,
MaxChars(10).SetFont(SansSerifZ(16)).SetFrame(FieldFrame()).LeftPosZ(136,
64).TopPosZ(308, 19))

 STRING_ (CUSTNAME, 150) NOT_NULL
 STRING_ (EMAIL, 150)
 STRING_ (CONTACT, 150)
 STRING_ (ADDRESS, 150)
```

```
 STRING_ (CITY, 50)
 STRING_ (STATE, 20)
 STRING_ (ZIP, 10)
 INT_ (TAXABLE) NOT_NULL
*/

AddCustomer::AddCustomer()
{
 CtrlLayoutOKCancel(*this, "Add Customer");
 ctrls
  (CUSTNAME, txtCustName)
  (EMAIL, txtCustEmail)
  (CONTACT, txtCustPhone)
  (ADDRESS, txtCustAddress)
  (CITY, txtCustCity)
  (STATE, txtCustState)
  (ZIP, txtCustZip)
  // (TAXABLE, chkTaxable)
 ;
}


CustomersWindow::CustomersWindow() {


 CtrlLayout(*this, "Customers");
 /*
 btnAddCustomer << [=] { btnAddCustomerClick(); }; // THISBACK is not needed in c++11 world
and could be replaced with lambda.
 btnSearchCustomer << [=] { btnSearchCustomerClick(); }; // assisted by forum user Klugier
 btnUpdateCustomer << [=] { btnUpdateCustomerClick(); };
 */

 if(FileExists(myConfig.configfile))
 {
     VectorMap<String, String> cfg = LoadIniFile(myConfig.configfile);
     myConfig.DBFile = cfg.Get("DBFile", Null);
 }
 else {
 myConfig.DBFile = myConfig.SelectDB();
 }

 Sqlite3Session sqlite3;
 if(!sqlite3.Open(myConfig.DBFile)) {
 Exclamation("Can't create or open database file\n");
 return;
 }
```

```
  SQL = sqlite3;
  /*
  Sql sql;
  */

  CustArray.SetTable(CUSTOMERS, CUST_ID);

  // CustArray.Join(BOOK_ID, book); // joins id from other db to this id
  CustArray.AddColumn(CUSTNAME, "Name"); // .SetConvert(DateIntConvert());
  CustArray.AddColumn(EMAIL, "Email"); // .SetConvert(DateIntConvert());
  CustArray.AddColumn(CONTACT, "Phone");
  CustArray.AddColumn(ADDRESS, "Address");
  CustArray.AddColumn(CITY, "City");
  CustArray.AddColumn(STATE, "State");
  CustArray.AddColumn(ZIP, "Zip");
  CustArray.AddColumn(TAXABLE, "Taxable?");
  CustArray.ColumnWidths("40 40 20 50 20 15 10 5");
  CustArray.Appending().NoRemoving();
  CustArray.SetOrderBy(CUST_ID);

  CustArray.Query();

  CustArray.WhenLeftDouble = [=] { EditRow(); };

}

void CustomersWindow::EditRow()
{
 int idNum;
 if(!CustArray.IsCursor())
  return;
 idNum = CustArray.GetKey();
 AddCustomer dlg;

 SQL * Select(CUST_ID).From(CUSTOMERS).Where(CUST_ID == idNum);
 if(!dlg.ctrls.Fetch(SQL))
  return;
 if(dlg.Run() != IDOK)
  return;
 SQL * dlg.ctrls.Update(CUSTOMERS).Where(CUST_ID == idNum);
 CustArray.Query();
 CustArray.FindSetCursor(idNum);
}
/*
void CustomersWindow::btnAddCustomerClick()
{

 PromptOK(__func__);
```

```
}

void CustomersWindow::btnUpdateCustomerClick()
{
 PromptOK(__func__);
}

void CustomersWindow::btnSearchCustomerClick()
{
 PromptOK(__func__);
}
*/
```

Now, when I run this, it gets to line 109:
SQL * Select(CUST_ID).From(CUSTOMERS).Where(CUST_ID == idNum);

And gives me an invalid memory access with:

 :?:

File Attachments
1) latest.png, downloaded 712 times