Subject: U++ needs wizards
Posted by lectus on Mon, 25 Jun 2012 23:36:37 GMT
View Forum Message <> Reply to Message

Hello!
Here is a feature request that should be easy enough to implement and would attract more new users and make our lives easier:

U++ should have project creation wizards!

Yes, I know there are ready project templates when you create a package.

But I mean more elaborated wizards. For example, when you create a new database app it asks a few questions like:
* Database type: sqlite, mysql, etc
* Database connection string (or file name for sqlite): test.db
* Do you want a default SqlArray with basic CRUD operations? Yes / No
* Define your tables (in a dialog)
* Finish

People coming from other IDEs love wizards!

Subject: Re: U++ needs wizards
Posted by dolik.rce on Tue, 26 Jun 2012 05:17:44 GMT
View Forum Message <> Reply to Message

lectus wrote on Tue, 26 June 2012 01:36But I mean more elaborated wizards. For example, when you create a new database app it asks a few questions like:
* Database type: sqlite, mysql, etc
* Database connection string (or file name for sqlite): test.db
* Do you want a default SqlArray with basic CRUD operations? Yes / No
* Define your tables (in a dialog)
* Finish
Hi lectus,

Out of the example you posted, 80% can be already be done very simply using templates The only part that is not possible right now is opening a dialog to specify more details, but it could be probably added to the templates too. (You're free to try it)

I personally don't like wizards that much, IMHO having all the setting in one dialog (as in the create new package dialog) is much more efficient. OTOH the only wizard currently present in theide is my work I understand that some users like it and that it sometimes makes things easier for beginners.

BTW: If you are interested in wizards and want to try to code some (e.g. a mock-up or example implementation to convince everyone that wizards are cool), there is a ready made class in

uppsrc/ide/SrcUpdater/Wizard.h just for that.

Best regards,
Honza

Subject: Re: U++ needs wizards
Posted by sergeynikitin on Fri, 14 Sep 2012 08:31:55 GMT
View Forum Message <> Reply to Message

Wizard not needed. We have classes.
For example.
I use my own class for some dictionaries with CRUD and RepGen calls and some other futures.
I write to describe one of dictionaries very small code like this:

```
//==dict_little_dicts.h=========================================================
=======//
class DictTypeOfCar : public DictStd {
public:
 typedef DictTypeOfCar CLASSNAME;

 DictTypeOfCar();
};
//==dict_little_dicts.cpp=======================================================
======//
DictTypeOfCar::DictTypeOfCar(){
 window_title = "Car type";
 ReportClass("Print car type");
 ReportListClass("Print list of car type");
 sqltable = TYPEOFCAR;
 sqlid = TOC_ID; sqlname = TOC_NAME;
 colname_id = "Id"; colname_name = "Type Name";
 HelpTopic("Dictionaries#TypeOfCar");

 DictStdSetup();
}
//==============================================================================
================//
```

Or, if I need some additional functionality:

```
//==dict_little_dicts.h=========================================================
=======//
class DictStatus : public DictStd {
public:
```

```
 typedef DictStatus CLASSNAME;
 ColorCtrl STA_Color;
 DictStatus();
 void virtual AddingColumns();
 void virtual LoadOneRecord();
};
//==dict_little_dicts.cpp=========================================================
======//
DictStatus::DictStatus(){
 window_title = "Statuses";
 ReportClass("Print status");
 ReportListClass("Print status list");
 sqltable = STATUS;
 sqlid = STA_ID; sqlname = STA_NAME;
 colname_id = "Id"; colname_name = "Status";
 HelpTopic("Dictionaries#Status");
 DictStdSetup();
}

void DictStatus::AddingColumns() {
 list.AddIndex(STA_ID);
 list.AddColumn(STA_NAME,"Status name",100).Edit(DctName);
 list.AddColumn(STA_COLOR,"Color",50).SetDisplay(Single<DisplayColorRect>()).Edit(STA_Col
or);
 STA_Color.SetConvert(Single<ConvertColor>());
 stdctrls
  (sqlname,DctName)
  (STA_COLOR,STA_Color)
 ;
 STA_Color.SetConvert(Single<ConvertColor>());
}

void DictStatus::LoadOneRecord(){
 list(STA_ID) = Dict::sqld[STA_ID];
 list(STA_NAME) = Dict::sqld[STA_NAME];
 list(STA_COLOR) = Dict::sqld[STA_COLOR];
}

//=================================================================================
================//
```

You can see, with wizard you spent more more time.
Classes help write ONE feature with ONE code string.

Usually in application 10-30 dictionaries and 3-10 main data tables.

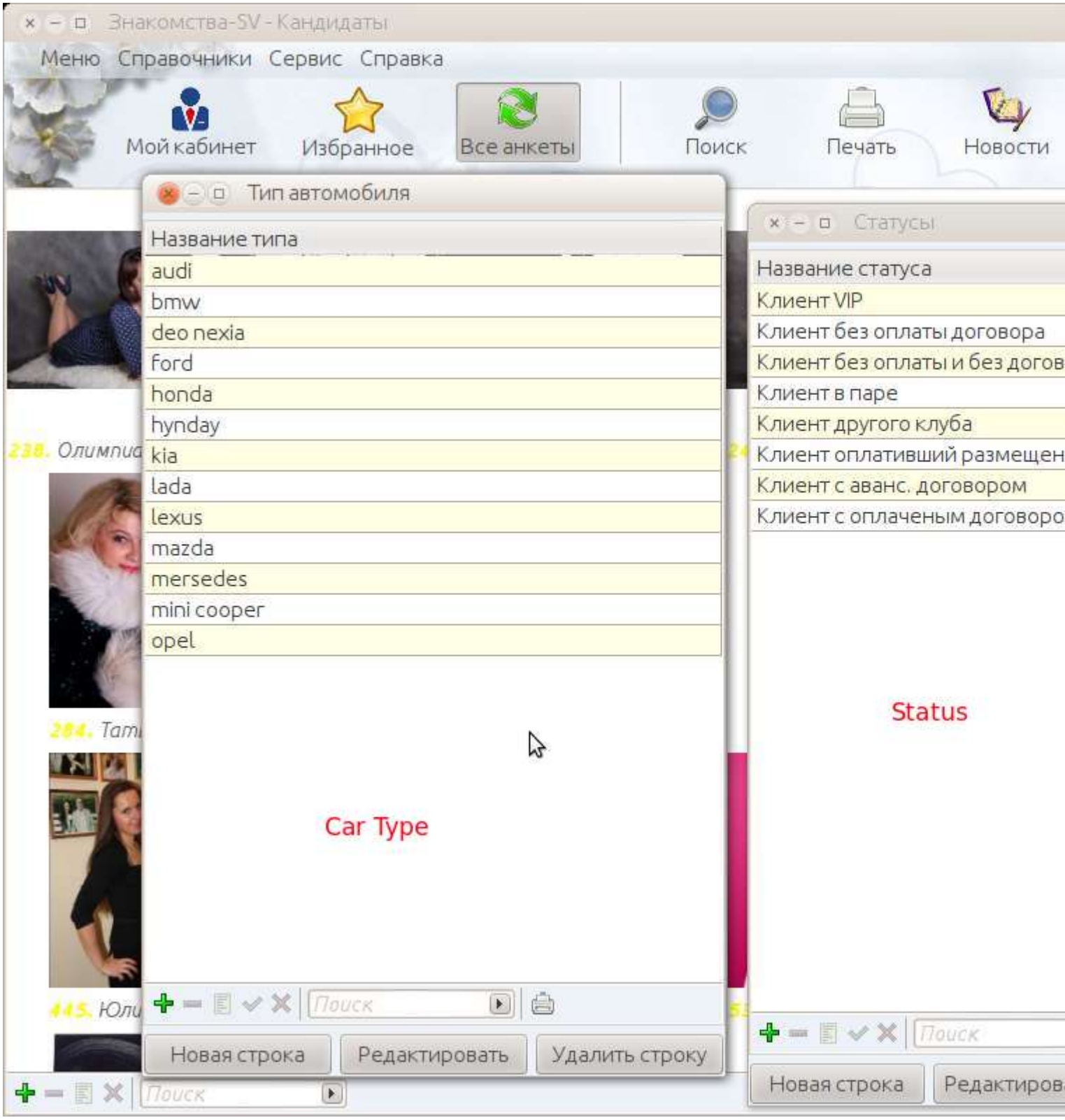
Make classes for this 2 things and other will be very quckly.

When I start write next application I just copy DictStd call and little rewrite main table CRUD classes. And application is ready.

The result is two beautiful DictWindow:

File Attachments

1) Çíàêîìñòâà-SV - Êàíäèäàòû_007.jpeg , downloaded 1027 times



Subject: Re: U++ needs wizards
Posted by sergeynikitin on Fri, 14 Sep 2012 14:39:45 GMT

IDEA!!!
Add simple wizard for template-driven code generation of Dict's or some other Simple window.

Maybe it will be not in Assist menu or not only in Assist. Maybe in File menu add item "Add with wizard" for newbies.