Лабораторна робота №5

Тема: Структури даних(масиви, записи).

Студента групи 141

Бужака Андрія

Варіант №4

Завдання: Скласти програму, яка

1. реалізує інтерфейс користувача при роботі з даними запису

2. додає записи в масив

3. виводить записи на екран

4. здійснює пошук запису за значенням обраних полів

5. видаляє зазначений запис із масиву

Функціонально закінчені модулі оформити у вигляді процедур.

Код програми:

**Файл UMain.cpp**

//---------------------------------------------------------------------------

#include <vcl.h>

#pragma hdrstop

#include "UMain.h"

#include "USearch.h"

#include "URandomContent.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

//---------------------------------------------------------------------------

TAvto arrAvto[1000];

int N=0;

int idAvto=0;

void FNew()

{

if(N<1000)

{

arrAvto[N].id=N;

idAvto=N;

}

else

ShowMessage("Переповнення масиву!");

}

TAvto FGet()

{

return arrAvto[idAvto];

}

void FSave(String num,String mrk,String nl,String nf,String as,String ah,String af,String rcpt)

{

arrAvto[idAvto].number=num;

arrAvto[idAvto].mark=mrk;

arrAvto[idAvto].name.last=nl;

arrAvto[idAvto].name.first=nf;

arrAvto[idAvto].address.street=as;

arrAvto[idAvto].address.house=ah;

arrAvto[idAvto].address.flat=af;

arrAvto[idAvto].receipt=rcpt;

N++;

}

void FDelete()

{

N--;

for(int i=idAvto;i<N;i++)

arrAvto[i]=arrAvto[i+1];

if(idAvto==N)

idAvto--;

}

void FNext()

{

if(idAvto<N-1)

idAvto++;

}

void FPrev()

{

if(idAvto>0)

idAvto--;

}

void FFirst()

{

if(idAvto>0)

idAvto=0;

}

void FLast()

{

idAvto=N-1;

}

void FSetId(int id)

{

idAvto=0;

while(idAvto<N && arrAvto[idAvto].id!=id)

idAvto++;

}

void FSearch(String s,TAvto arrRez[],int &k)

{

k=0;

for(int i=0;i<N;i++)

{

if(arrAvto[i].number.Pos(s)>0 ||

arrAvto[i].mark.Pos(s)>0 ||

arrAvto[i].name.last.Pos(s)>0 ||

arrAvto[i].name.first.Pos(s)>0 ||

arrAvto[i].address.street.Pos(s)>0 ||

arrAvto[i].address.house.Pos(s)>0 ||

arrAvto[i].address.flat.Pos(s)>0 ||

arrAvto[i].receipt.Pos(s)>0

)

{

arrRez[k]=arrAvto[i];

k++;

}

}

}

//---------------------------------------------------------------------------

TfrmMain \*frmMain;

//---------------------------------------------------------------------------

\_\_fastcall TfrmMain::TfrmMain(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void enabledOnOff(bool f)

{

frmMain->edtNumber->Enabled=f;

frmMain->edtMark->Enabled=f;

frmMain->edtNameLast->Enabled=f;

frmMain->edtNameFirst->Enabled=f;

frmMain->edtAddressStreet->Enabled=f;

frmMain->edtAddressHouse->Enabled=f;

frmMain->edtAddressFlat->Enabled=f;

frmMain->edtReceipt->Enabled=f;

frmMain->btnFirst->Enabled=!f;

frmMain->btnPrev->Enabled=!f;

frmMain->btnNext->Enabled=!f;

frmMain->btnLast->Enabled=!f;

frmMain->btnNew->Enabled=!f;

frmMain->btnUpdate->Enabled=!f;

frmMain->btnDelete->Enabled=!f;

frmMain->btnSave->Enabled=f;

frmMain->btnRandom->Enabled=f;

}

void FShow(TAvto avto)

{

frmMain->edtNumber->Text=avto.number;

frmMain->edtMark->Text=avto.mark;

frmMain->edtNameLast->Text=avto.name.last;

frmMain->edtNameFirst->Text=avto.name.first;

frmMain->edtAddressStreet->Text=avto.address.street;

frmMain->edtAddressHouse->Text=avto.address.house;

frmMain->edtAddressFlat->Text=avto.address.flat;

frmMain->edtReceipt->Text=avto.receipt;

}

void enabledStep()

{

if(N>1)

{

if(idAvto==0)

{

frmMain->btnFirst->Enabled=false;

frmMain->btnPrev->Enabled=false;

frmMain->btnNext->Enabled=true;

frmMain->btnLast->Enabled=true;

}

if(idAvto==N-1)

{

frmMain->btnFirst->Enabled=true;

frmMain->btnPrev->Enabled=true;

frmMain->btnNext->Enabled=false;

frmMain->btnLast->Enabled=false;

}

if(idAvto>0 && idAvto<N-1)

{

frmMain->btnFirst->Enabled=true;

frmMain->btnPrev->Enabled=true;

frmMain->btnNext->Enabled=true;

frmMain->btnLast->Enabled=true;

}

}

else

{

frmMain->btnFirst->Enabled=false;

frmMain->btnPrev->Enabled=false;

frmMain->btnNext->Enabled=false;

frmMain->btnLast->Enabled=false;

}

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmMain::btnNewClick(TObject \*Sender)

{

enabledOnOff(true);

frmMain->edtNumber->Text="";

frmMain->edtMark->Text="";

frmMain->edtNameLast->Text="";

frmMain->edtNameFirst->Text="";

frmMain->edtAddressStreet->Text="";

frmMain->edtAddressHouse->Text="";

frmMain->edtAddressFlat->Text="";

frmMain->edtReceipt->Text="";

FNew();

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmMain::btnUpdateClick(TObject \*Sender)

{

N--;

enabledOnOff(true);

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmMain::btnSaveClick(TObject \*Sender)

{

FSave(edtNumber->Text,edtMark->Text,edtNameLast->Text,edtNameFirst->Text,edtAddressStreet->Text,edtAddressHouse->Text,edtAddressFlat->Text,edtReceipt->Text);

enabledOnOff(false);

enabledStep();

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmMain::btnDeleteClick(TObject \*Sender)

{

if(Application->MessageBox("Ви дійсно бажаєте видалити запис?","Видалення запису",MB\_YESNO)==mrYes)

{

if(N>1)

{

FDelete();

FShow(FGet());

}

else

{

FDelete();

enabledOnOff(true);

frmMain->edtNumber->Text="";

frmMain->edtMark->Text="";

frmMain->edtNameLast->Text="";

frmMain->edtNameFirst->Text="";

frmMain->edtAddressStreet->Text="";

frmMain->edtAddressHouse->Text="";

frmMain->edtAddressFlat->Text="";

frmMain->edtReceipt->Text="";

FNew();

}

}

enabledStep();

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmMain::FormCreate(TObject \*Sender)

{

enabledOnOff(true);

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmMain::btnFirstClick(TObject \*Sender)

{

FFirst();

FShow(FGet());

enabledStep();

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmMain::btnPrevClick(TObject \*Sender)

{

FPrev();

FShow(FGet());

enabledStep();

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmMain::btnNextClick(TObject \*Sender)

{

FNext();

FShow(FGet());

enabledStep();

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmMain::btnLastClick(TObject \*Sender)

{

FLast();

FShow(FGet());

enabledStep();

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmMain::btnSearchClick(TObject \*Sender)

{

frmSearch->Show();

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmMain::btnRandomClick(TObject \*Sender)

{

char s[100];

FRandom1(s);

frmMain->edtNumber->Text=s;

FRandom2(s);

frmMain->edtMark->Text=s;

FRandom3(s);

frmMain->edtNameLast->Text=s;

FRandom4(s);

frmMain->edtNameFirst->Text=s;

FRandom5(s);

frmMain->edtAddressStreet->Text=s;

FRandom6(s);

frmMain->edtAddressHouse->Text=s;

FRandom7(s);

frmMain->edtAddressFlat->Text=s;

FRandom8(s);

frmMain->edtReceipt->Text=s;

}

//---------------------------------------------------------------------------

**Файл UMain.h**

//---------------------------------------------------------------------------

#ifndef UMainH

#define UMainH

//---------------------------------------------------------------------------

#include <Classes.hpp>

#include <Controls.hpp>

#include <StdCtrls.hpp>

#include <Forms.hpp>

#include <String.h>

//---------------------------------------------------------------------------

struct TName

{

String last;

String first;

};

struct TAddress

{

String street;

String house;

String flat;

};

struct TAvto

{

int id;

String number;

String mark;

TName name;

TAddress address;

String receipt;

};

void FShow(TAvto);

void FNew();

TAvto FGet();

void FSave(String num,String mrk,String nl,String nf,String as,String ah,String af,String rcpt);

void FDelete();

void FNext();

void FPrev();

void FFirst();

void FLast();

void FSetId(int);

void FSearch(String s,TAvto arrRez[],int &k);

//---------------------------------------------------------------------------

class TfrmMain : public TForm

{

\_\_published: // IDE-managed Components

TEdit \*edtNumber;

TEdit \*edtMark;

TEdit \*edtNameLast;

TEdit \*edtNameFirst;

TEdit \*edtAddressStreet;

TButton \*btnNew;

TButton \*btnFirst;

TButton \*btnPrev;

TButton \*btnUpdate;

TButton \*btnDelete;

TButton \*btnSearch;

TButton \*btnNext;

TButton \*btnSave;

TButton \*btnLast;

TEdit \*edtAddressHouse;

TEdit \*edtAddressFlat;

TEdit \*edtReceipt;

TLabel \*Label1;

TLabel \*Label2;

TLabel \*Label3;

TLabel \*Label4;

TLabel \*Label5;

TLabel \*Label6;

TLabel \*Label7;

TLabel \*Label8;

TLabel \*Label9;

TButton \*btnRandom;

void \_\_fastcall btnNewClick(TObject \*Sender);

void \_\_fastcall btnUpdateClick(TObject \*Sender);

void \_\_fastcall btnSaveClick(TObject \*Sender);

void \_\_fastcall FormCreate(TObject \*Sender);

void \_\_fastcall btnFirstClick(TObject \*Sender);

void \_\_fastcall btnPrevClick(TObject \*Sender);

void \_\_fastcall btnNextClick(TObject \*Sender);

void \_\_fastcall btnLastClick(TObject \*Sender);

void \_\_fastcall btnDeleteClick(TObject \*Sender);

void \_\_fastcall btnSearchClick(TObject \*Sender);

void \_\_fastcall btnRandomClick(TObject \*Sender);

private: // User declarations

public: // User declarations

\_\_fastcall TfrmMain(TComponent\* Owner);

};

//---------------------------------------------------------------------------

extern PACKAGE TfrmMain \*frmMain;

//---------------------------------------------------------------------------

#endif

**Файл USearch.cpp**

//---------------------------------------------------------------------------

#include <vcl.h>

#pragma hdrstop

#include "USearch.h"

#include "UMain.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

TfrmSearch \*frmSearch;

TAvto arrRez[1000];

int nRez=0;

//---------------------------------------------------------------------------

\_\_fastcall TfrmSearch::TfrmSearch(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmSearch::FormCreate(TObject \*Sender)

{

stgSearch->Cells[0][0]="Номер автомобіля";

stgSearch->Cells[1][0]="Марка";

stgSearch->Cells[2][0]="Прізвище власника";

stgSearch->Cells[3][0]="Ім'я власника";

stgSearch->Cells[4][0]="Вулиця";

stgSearch->Cells[5][0]="Будинок";

stgSearch->Cells[6][0]="Квартира";

stgSearch->Cells[7][0]="Штрафи";

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmSearch::edtSearchChange(TObject \*Sender)

{

FSearch(edtSearch->Text,arrRez,nRez);

stgSearch->RowCount=nRez+1;

for(int i=0;i<nRez;i++)

{

stgSearch->Cells[0][i+1]=arrRez[i].number;

stgSearch->Cells[1][i+1]=arrRez[i].mark;

stgSearch->Cells[2][i+1]=arrRez[i].name.last;

stgSearch->Cells[3][i+1]=arrRez[i].name.first;

stgSearch->Cells[4][i+1]=arrRez[i].address.street;

stgSearch->Cells[5][i+1]=arrRez[i].address.house;

stgSearch->Cells[6][i+1]=arrRez[i].address.flat;

stgSearch->Cells[7][i+1]=arrRez[i].receipt;

}

if(stgSearch->RowCount>1)

stgSearch->FixedRows=1;

}

//---------------------------------------------------------------------------

void \_\_fastcall TfrmSearch::stgSearchSelectCell(TObject \*Sender, int ACol,

int ARow, bool &CanSelect)

{

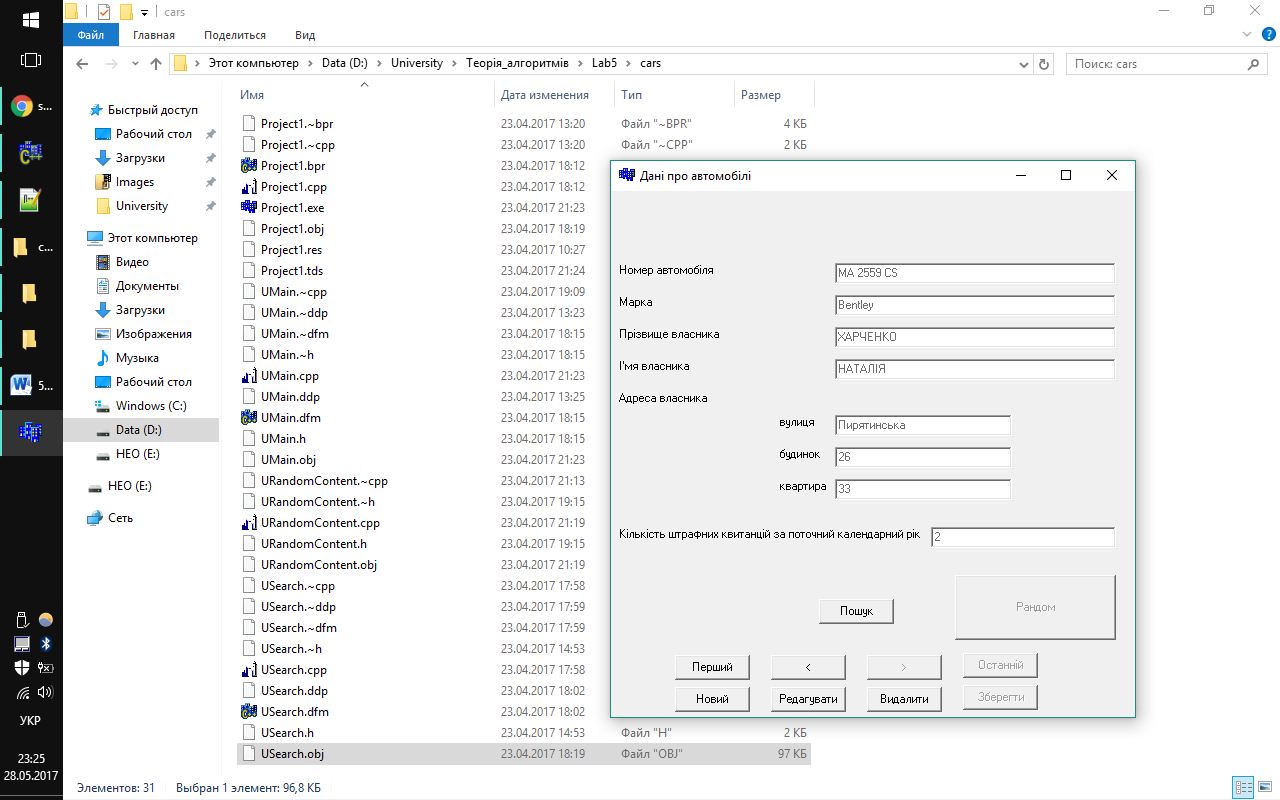
FSetId(arrRez[ARow-1].id);

FShow(FGet());

}

//---------------------------------------------------------------------------

**Головна форма:**



**Форма пошуку:**

