Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №9**

**«Работа с типизированными файлами»**

**МДК 05.02 «РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ»**

Выполнил: студент учебной группы

ИСПк-203-52-00

Чарушин Егор Вадимович

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2024

1. **Цель работы:**

Получить представления об организации хранения и использования информации посредством типизированных файлов, закрепить навыки создания пользовательского интерфейса.

1. **Формулировка задания:**
2. Сформировать структуру записи, с которой будет осуществляться работа;
3. Разработать приложения, позволяющее извлекать набор записей из типизированного файла, визуализировать данный набор, а также позволяющее изменять данных и выполнять сохранение проделанных изменений.
4. **Код программы:**

unit Main;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, Forms, Controls, Graphics, Dialogs, ExtCtrls,

Buttons, edit, Grids;

type

Contacts = record

Name: string[100]; // измените на Array[0..99] of Char

Country: string[20]; // измените на Array[0..19] of Char

Speed: integer;

Color: string[20];

Cost: double;

Note: string[20]; // изменено на string

end; //record

type

{ TfMain }

TfMain = class(TForm)

Panel1: TPanel;

bAdd: TSpeedButton;

bEdit: TSpeedButton;

bDel: TSpeedButton;

bSort: TSpeedButton;

SG: TStringGrid;

procedure bAddClick(Sender: TObject);

procedure bDelClick(Sender: TObject);

procedure bEditClick(Sender: TObject);

procedure bSortClick(Sender: TObject);

procedure FormClose(Sender: TObject; var CloseAction: TCloseAction);

procedure FormCreate(Sender: TObject);

private

public

end;

var

fMain: TfMain;

adres: string; //адрес, откуда запущена программа

implementation

{$R \*.lfm}

{ TfMain }

procedure TfMain.bAddClick(Sender: TObject);

var

NoteValue: boolean;

CostValue: double;

begin

fEdit.eName.Text:= '';

fEdit.eCountry.Text:= '';

fEdit.eSpeed.Text:= '';

fEdit.eColor.Text:= '';

fEdit.eCost.Text:= '';

fEdit.ModalResult:= mrNone;

fEdit.ShowModal;

if (fEdit.eName.Text= '') or (fEdit.eCountry.Text= '') then exit;

if fEdit.ModalResult <> mrOk then exit;

SG.RowCount:= SG.RowCount + 1;

SG.Cells[0, SG.RowCount-1]:= fEdit.eName.Text;

SG.Cells[1, SG.RowCount-1]:= fEdit.eCountry.Text;

SG.Cells[2, SG.RowCount-1]:= fEdit.eSpeed.Text;

SG.Cells[3, SG.RowCount-1]:= fEdit.eColor.Text;

SG.Cells[4, SG.RowCount-1]:= fEdit.eCost.Text;

// Преобразуем текст из ComboBox в строку

SG.Cells[5, SG.RowCount-1] := fEdit.CBNote.Text;

end;

procedure TfMain.bDelClick(Sender: TObject);

begin

//если данных нет - выходим:

if SG.RowCount = 1 then exit;

//иначе выводим запрос на подтверждение:

if MessageDlg('Требуется подтверждение',

'Вы действительно хотите удалить автомобиль "' +

SG.Cells[0, SG.Row] + '"?',

mtConfirmation, [mbYes, mbNo, mbIgnore], 0) = mrYes then

SG.DeleteRow(SG.Row);

end;

procedure TfMain.bEditClick(Sender: TObject);

begin

if SG.RowCount = 1 then exit;

fEdit.eName.Text:= SG.Cells[0, SG.Row];

fEdit.eCountry.Text:= SG.Cells[1, SG.Row];

fEdit.eSpeed.Text:= SG.Cells[2, SG.Row];

fEdit.eColor.Text:= SG.Cells[3, SG.Row];

fEdit.eCost.Text:= SG.Cells[4, SG.Row];

fEdit.CBNote.Text := SG.Cells[5, SG.Row];

fEdit.ModalResult:= mrNone;

fEdit.ShowModal;

if fEdit.ModalResult = mrOk then begin

SG.Cells[0, SG.Row]:= fEdit.eName.Text;

SG.Cells[1, SG.Row]:= fEdit.eCountry.Text;

SG.Cells[2, SG.Row]:= fEdit.eSpeed.Text;

SG.Cells[3, SG.Row]:= fEdit.eColor.Text;

SG.Cells[4, SG.Row]:= fEdit.eCost.Text;

SG.Cells[5, SG.Row]:= fEdit.CBNote.Text;

end;

end;

procedure TfMain.bSortClick(Sender: TObject);

begin

//если данных в сетке нет - просто выходим:

if SG.RowCount = 1 then exit;

//иначе сортируем список:

SG.SortColRow(true, 0);

end;

procedure TfMain.FormClose(Sender: TObject; var CloseAction: TCloseAction);

var

MyCar: Contacts; // для очередной записи

f: file of Contacts; // файл данных

i: integer; // счетчик цикла

begin

// если строки данных пусты, просто выходим

if SG.RowCount = 1 then exit;

// иначе открываем файл для записи

try

AssignFile(f, adres + 'cars.dat');

Rewrite(f);

// теперь цикл - от первой до последней записи сетки

for i := 1 to SG.RowCount - 1 do

begin

// получаем данные текущей записи

MyCar.Name := SG.Cells[0, i];

MyCar.Country := SG.Cells[1, i];

MyCar.Speed := StrToInt(SG.Cells[2, i]); // преобразуем строку в целое число

MyCar.Color := SG.Cells[3, i];

MyCar.Cost := StrToFloat(SG.Cells[4, i]); // преобразуем строку в число с плавающей точкой

MyCar.Note := SG.Cells[5, i]; // сохраняем строку

// записываем данные

Write(f, MyCar);

end;

finally

CloseFile(f);

end;

end;

procedure TfMain.FormCreate(Sender: TObject);

var

MyCont: Contacts; //для очередной записи

f: file of Contacts; //файл данных

begin

//сначала получим адрес программы:

adres:= ExtractFilePath(ParamStr(0));

//настроим сетку:

SG.Cells[0, 0]:= 'Название';

SG.Cells[1, 0]:= 'Страна';

SG.Cells[2, 0]:= 'Мощность, лс';

SG.Cells[3, 0]:= 'Цвет';

SG.Cells[4, 0]:= 'Стоимость, млн. руб';

SG.Cells[5, 0]:= 'В наличии';

SG.ColWidths[0]:= 150;

SG.ColWidths[1]:= 150;

SG.ColWidths[2]:= 150;

SG.ColWidths[3]:= 150;

SG.ColWidths[4]:= 150;

SG.ColWidths[5]:= 150;

//если файла данных нет, просто выходим:

if not FileExists(adres + 'cars.dat') then exit;

//иначе файл есть, открываем его для чтения и

//считываем данные в сетку:

try

AssignFile(f, adres + 'cars.dat');

Reset(f);

//теперь цикл - от первой до последней записи сетки:

while not Eof(f) do begin

//считываем новую запись:

Read(f, MyCont);

//добавляем в сетку новую строку, и заполняем её:

SG.RowCount := SG.RowCount + 1;

SG.Cells[0, SG.RowCount - 1] := MyCont.Name;

SG.Cells[1, SG.RowCount - 1] := MyCont.Country;

SG.Cells[2, SG.RowCount - 1] := IntToStr(MyCont.Speed); // Преобразование в строку

SG.Cells[3, SG.RowCount - 1] := MyCont.Color;

SG.Cells[4, SG.RowCount - 1] := FloatToStr(MyCont.Cost); // Преобразование в строку

SG.Cells[5, SG.RowCount - 1] := MyCont.Note;

end;

finally

CloseFile(f);

end;

end;

end.

end;

1. **Результат выполнения программы.**

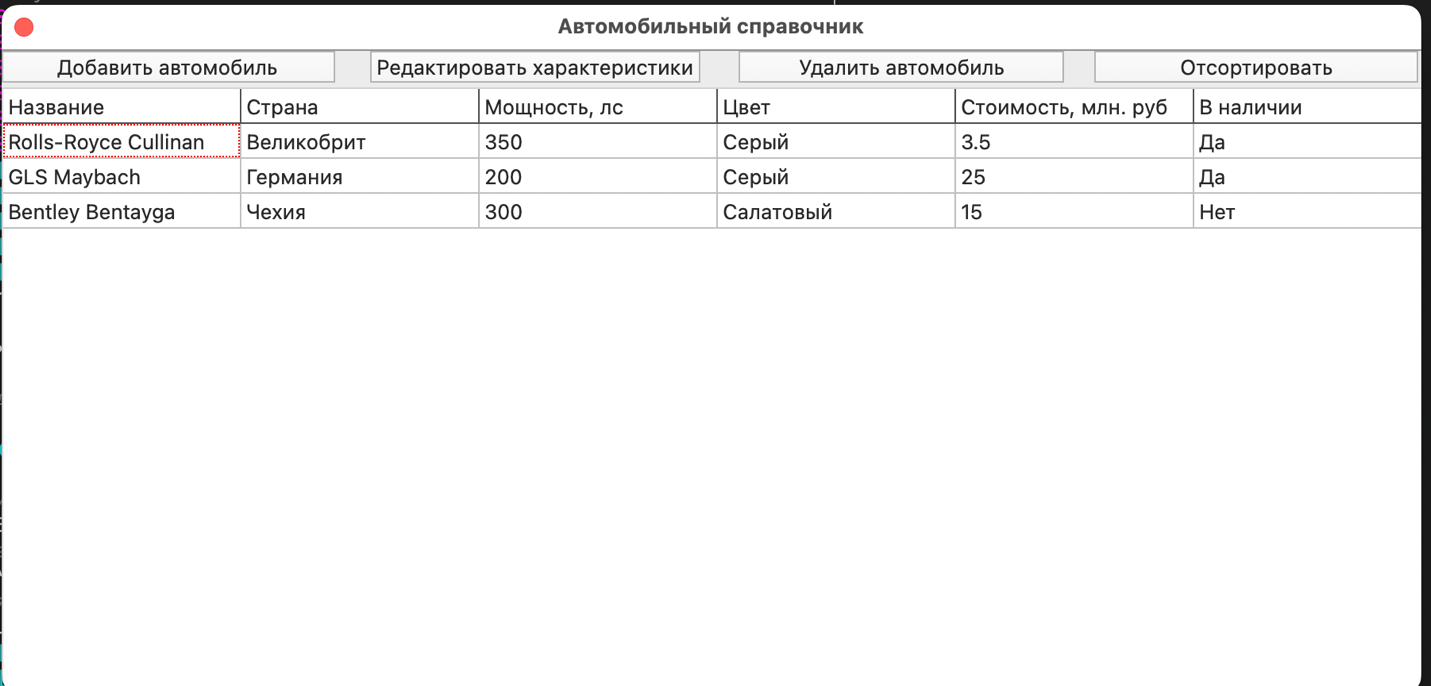


Рисунок 1 – Интерфейс главного окна.

Изображение выглядит как текст, электроника, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 2 – Интерфейс добавления автомобиля.

Изображение выглядит как текст, электроника, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 3 – Интерфейс редактирования.

Изображение выглядит как текст, снимок экрана, Мобильный телефон, гаджет

Автоматически созданное описание

Рисунок 4 – Интерфейс удаления.

1. **Вывод:**

В данной работе были проделаны операции по получению информации об организации хранения и использования информации посредством типизированных файлов.

Были успешно закреплены навыки создания пользовательского интерфейса, а также создано приложения, позволяющие добавлять записи об автомобилях, их редактировать, удалять и сортировать.

Таким образом все желаемые цели были успешно достигнуты, все задачи были выполнены в полном объеме и успешно закреплены.