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»**

**ПО «МДК 05.02 Разработка кода информационных систем»**

Выполнил: студент учебной группы

ИСПк-202-52-00

Сосновский Никита Владимирович

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2024

**Цель работы:** получить представление об организации хранения и использования информации посредством типизированных файлов, закрепить навыки создания пользовательского интерфейса.

**Задание**

1. Сформировать структуру записи, с которой будет осуществляться работа.

2. Разработать приложение, позволяющее извлекать набор записей из типизированного файла, визуализировать данный набор, а также позволяющее изменять данные и выполнять сохранение проделанных изменений.

**Описание алгоритма**

Элементы структуры записи:

1) Имя: string

2) Уровень: integer

3) Учитель: Boolean

4) Гильдия: string

5) Резиденция: string

Добавляем вторую форму, процедуры для обработки нажатий на кнопки, а также процедуры FormClose и FormCreate.

**Результат выполнения задания**

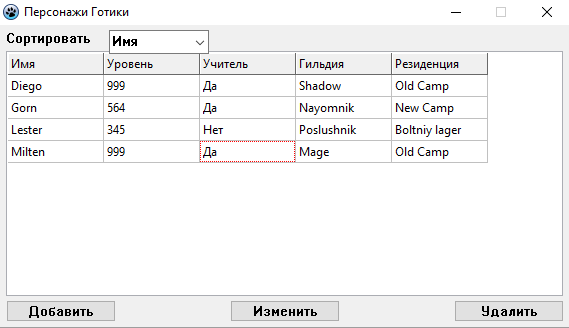


Рисунок 1 – Главное окно

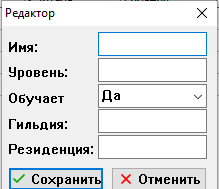


Рисунок 2 – Окно для добавления или изменения

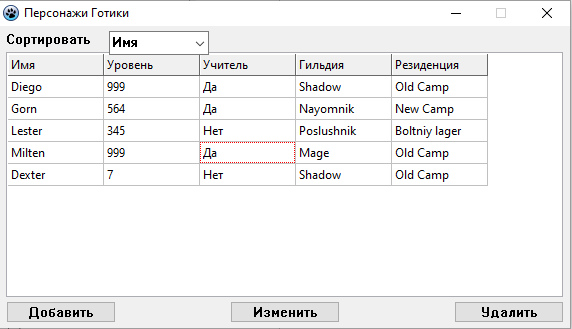


Рисунок 3 – Добавление данных о новом персонаже (Dexter)

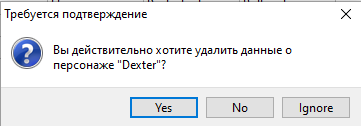


Рисунок 4 – Удаление данных о персонаже

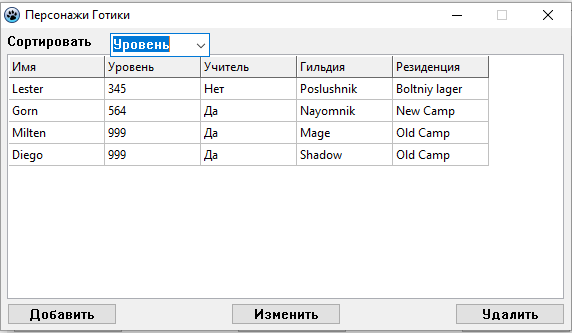


Рисунок 5 – Сортировка по уровню

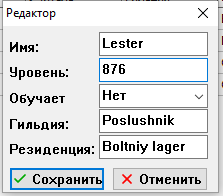


Рисунок 6 – Изменение данных о персонаже

**Исходный код**

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

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

Grids, StdCtrls, unit2;

type

{ TGothicCh }

TGothicCh = class(TForm)

CBsort: TComboBox;

lSort: TLabel;

bAdd: TSpeedButton;

bDel: TSpeedButton;

bEdit: TSpeedButton;

SG: TStringGrid;

procedure bAddClick(Sender: TObject);

procedure bDelClick(Sender: TObject);

procedure bEditClick(Sender: TObject);

procedure CBsortChange(Sender: TObject);

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

procedure FormCreate(Sender: TObject);

private

public

end;

type

Characters = record

name: string[100];

level: integer;

teach: boolean;

guild: string[100];

res: string[100];

end;

var

GothicCh: TGothicCh;

adresss: string;

implementation

{$R \*.lfm}

{ TGothicCh }

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

var Ch: Characters;

f: file of Characters;

i: integer;

begin

if SG.RowCount = 1 then exit;

try

AssignFile(f, adresss+'GothicChar.dat');

Rewrite(f);

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

Ch.name := SG.Cells[0, i];

Ch.level := StrToInt(SG.Cells[1, i]);

if SG.Cells[2, i] = 'Да' then

Ch.teach := True

else if SG.Cells[2, i] = 'Нет' then

Ch.teach := False;

Ch.guild := SG.Cells[3, i];

Ch.res := SG.Cells[4, i];

Write(f, Ch);

end;

finally

CloseFile(f);

end;

end;

procedure TGothicCh.FormCreate(Sender: TObject);

var

Ch: Characters;

f: file of Characters;

i: integer;

begin

adresss := ExtractFilePath(ParamStr(0));

SG.Cells[0, 0] := 'Имя';

SG.Cells[1, 0] := 'Уровень';

SG.Cells[2, 0] := 'Учитель';

SG.Cells[3, 0] := 'Гильдия';

SG.Cells[4, 0] := 'Резиденция';

for i := 0 to 4 do

SG.ColWidths[i] := 120;

if not FileExists(adresss+'GothicChar.dat') then exit;

try

AssignFile(f, adresss+'GothicChar.dat');

Reset(f);

while not eof(f) do begin

Read(f, Ch);

SG.RowCount := SG.RowCount + 1;

SG.Cells[0, SG.RowCount-1] := Ch.name;

SG.Cells[1, SG.RowCount-1] := IntToStr(Ch.level);

if Ch.teach then

SG.Cells[2, SG.RowCount-1] := 'Да'

else

SG.Cells[2, SG.RowCount-1] := 'Нет';

SG.Cells[3, SG.RowCount-1] := Ch.guild;

SG.Cells[4, SG.RowCount-1] := Ch.res;

end;

SG.SortColRow(true, 0);

finally

CloseFile(f);

end;

end;

procedure TGothicCh.CBsortChange(Sender: TObject);

begin

case CBSort.ItemIndex of

0: SG.SortColRow(true, 0);

1: SG.SortColRow(true, 1);

2: SG.SortColRow(true, 2);

3: SG.SortColRow(true, 3);

4: SG.SortColRow(true, 4);

end;

end;

procedure TGothicCh.bAddClick(Sender: TObject);

begin

fEdit.eName.Text := '';

fEdit.eLevel.Text := '';

fEdit.eGuild.Text := '';

fEdit.eRes.Text := '';

fEdit.ModalResult := mrNone;

fEdit.ShowModal;

if (fEdit.eName.Text = '') or (fEdit.eLevel.text = '') or (fEdit.eGuild.text = '') or (fEdit.eRes.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.eLevel.text;

SG.Cells[2, SG.RowCount - 1] := fEdit.CBteach.text;

SG.Cells[3, SG.RowCount - 1] := fEdit.eGuild.text;

SG.Cells[4, SG.RowCount - 1] := fEdit.eRes.text;

end;

procedure TGothicCh.bDelClick(Sender: TObject);

begin

if SG.RowCount = 1 then Exit;

if MessageDlg('Требуется подтверждение',

'Вы действительно хотите удалить данные о персонаже "' +

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

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

SG.DeleteRow(SG.Row);

end;

procedure TGothicCh.bEditClick(Sender: TObject);

begin

if SG.RowCount = 1 then exit;

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

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

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

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

fEdit.eRes.Text:= SG.Cells[4, 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.eLevel.Text;

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

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

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

end;

end;

end.

unit Unit2;

{$mode ObjFPC}{$H+}

interface

uses

Classes, SysUtils, Forms, Controls, Graphics, Dialogs, StdCtrls, Buttons;

type

{ TfEdit }

TfEdit = class(TForm)

BitBtn1: TBitBtn;

BitBtn2: TBitBtn;

CBteach: TComboBox;

eName: TEdit;

eGuild: TEdit;

eRes: TEdit;

eLevel: TEdit;

lName: TLabel;

lLevel: TLabel;

lTeach: TLabel;

lGuild: TLabel;

lRes: TLabel;

procedure FormShow(Sender: TObject);

private

public

end;

var

fEdit: TfEdit;

implementation

{$R \*.lfm}

procedure TfEdit.FormShow(Sender: TObject);

begin

eName.SetFocus;

end;

end.

**Вывод**

Таким образом, в результате выполнения работы было реализовано приложение, позволяющее извлекать набор записей из типизированного файла, визуализировать данный набор, а также позволяющее изменять данные и выполнять сохранение проделанных изменений.

Разработанная программа представляет собой пример работы с типизированными файлами.