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-202-52-00

Семаков Антон Константинович

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 для выполнения**

1. Сформировать структуру записи, с которой будет осуществляться работа, и согласовать её с преподавателем.
2. Разработать приложение, позволяющее извлекать набор записей из типизированного файла, визуализировать данный набор, а также позволяющее изменять данные и выполнять сохранение проделанных изменений.
3. В записи должно быть не менее 5 элементов, среди которых должно быть минимум 3 различных типа данных.

# **Описание алгоритма**

На главной форме расположены элементы управления для работы с записями философов:

Элементы интерфейса:

* **TBitBtn** - кнопки для добавления (**Add\_Note**), редактирования (**Edit\_Note**) и удаления (**Del\_Note**) записей.
* **TPanel** - панель для размещения кнопок управления.
* **TStringGrid** - таблица для отображения записей философов (**mRec**).

Основной алгоритм работы:

1. **Инициализация формы (FormCreate)**

При создании формы инициализируются элементы управления и заполняется заголовок таблицы.

Если файл данных существует, считываются данные из файла и заполняется таблица.

1. **Добавление записи (Add\_NoteClick)**

Очищаются поля ввода формы **createRecord**.

Отображается форма для ввода новой записи.

Если введены все необходимые данные и нажата кнопка "Сохранить", новая запись добавляется в таблицу.

1. **Удаление записи (Del\_NoteClick)**

Если в таблице есть данные, отображается диалог подтверждения удаления.

При подтверждении выбранная запись удаляется из таблицы.

1. **Редактирование записи (Edit\_NoteClick)**

Если в таблице есть данные, выбранная запись копируется в поля формы **createRecord**.

Отображается форма для редактирования записи.

При сохранении изменений запись обновляется в таблице.

1. **Закрытие формы (FormClose)**

Если в таблице есть данные, они записываются в файл при закрытии формы.

# **Код программы**

unit main\_module;

{$mode objfpc}{$H+}

interface

uses

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

Grids, ComCtrls, createRec;

type

{ TmForm }

TmForm = class(TForm)

Add\_Note: TBitBtn;

Edit\_Note: TBitBtn;

Del\_Note: TBitBtn;

Panel: TPanel;

mRec: TStringGrid;

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

procedure FormCreate(Sender: TObject);

procedure Add\_NoteClick(Sender: TObject);

procedure Del\_NoteClick(Sender: TObject);

procedure Edit\_NoteClick(Sender: TObject);

procedure PanelClick(Sender: TObject);

private

public

end;

type

Group = record

Name: string[100];

Year\_of\_foundation: integer;

Exist: boolean;

First\_release: integer;

Genre: string[30];

end;

var

mForm: TmForm;

adres: string;

implementation

{$R \*.lfm}

{ TmForm }

procedure TmForm.FormCreate(Sender: TObject);

var

new\_Group: Group; //для очередной записи

f: file of Group; //файл данных

i: integer; //счетчик цикла

begin

adres:= ExtractFilePath(ParamStr(0));

mRec.Cells[0, 0]:= 'Название группы';

mRec.Cells[1, 0]:= 'Год основания';

mRec.Cells[2, 0]:= 'Существует сейчас';

mRec.Cells[3, 0]:= 'первый релиз';

mRec.Cells[4, 0]:= 'Жанр';

//если файла данных нет, просто выходим:

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

//иначе файл есть, открываем его для чтения и

//считываем данные в сетку:

try

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

Reset(f);

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

while not Eof(f) do begin

//считываем новую запись:

Read(f, new\_Group);

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

mRec.RowCount:= mRec.RowCount + 1;

mRec.Cells[0, mRec.RowCount-1]:= new\_Group.Name;

mRec.Cells[1, mRec.RowCount-1]:= IntToStr(new\_Group.Year\_of\_foundation);

if (new\_Group.Exist) then mRec.Cells[2, mRec.RowCount-1]:= 'Да'

else mRec.Cells[2, mRec.RowCount-1]:='Нет';

mRec.Cells[3, mRec.RowCount-1]:= FloatToStr(new\_Group.First\_release);

mRec.Cells[4, mRec.RowCount-1]:= new\_Group.Genre;

end;

finally

CloseFile(f);

end;

end;

procedure TmForm.Add\_NoteClick(Sender: TObject);

begin

//очищаем поля, если там что-то есть:

createRecord.Name\_edit.Text:= '';

createRecord.Year\_of\_foundation\_edit.Text:= '';

createRecord.First\_release.Text:= '';

//устанавливаем ModalResult редактора в mrNone:

createRecord.ModalResult:= mrNone;

//теперь выводим форму:

createRecord.ShowModal;

//если пользователь ничего не ввел - выходим:

if (createRecord.Name\_edit.Text= '') or (createRecord.Year\_of\_foundation\_edit.Text= '') or (createRecord.First\_release.Text= '') then exit;

//если пользователь не нажал "Сохранить" - выходим:

if createRecord.ModalResult <> mrOk then exit;

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

mRec.RowCount:= mRec.RowCount + 1;

mRec.Cells[0, mRec.RowCount-1]:= createRecord.Name\_edit.Text;

mRec.Cells[1, mRec.RowCount-1]:= createRecord.Year\_of\_foundation\_edit.Text;

if (createRecord.Materialist\_chkbx.Checked) then mRec.Cells[2, mRec.RowCount-1]:= 'Да'

else mRec.Cells[2, mRec.RowCount-1]:='Нет';

mRec.Cells[3, mRec.RowCount-1]:= createRecord.First\_release.Text;

mRec.Cells[4, mRec.RowCount-1]:= createRecord.Dir\_of\_philosophy\_label\_list.Text;

end;

procedure TmForm.Del\_NoteClick(Sender: TObject);

begin

//если данных нет - выходим:

if mRec.RowCount = 1 then exit;

//иначе выводим запрос на подтверждение:

if MessageDlg('Требуется подтверждение',

'Вы действительно хотите удалить запись "' +

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

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

mRec.DeleteRow(mRec.Row);

end;

procedure TmForm.Edit\_NoteClick(Sender: TObject);

begin

//если данных в сетке нет - просто выходим:

if mRec.RowCount = 1 then exit;

//иначе записываем данные в форму редактора:

createRecord.Name\_edit.Text:= mRec.Cells[0, mRec.Row];

createRecord.Year\_of\_foundation\_edit.Text:= mRec.Cells[1, mRec.Row];

if (mRec.Cells[2, mRec.RowCount-1] = 'Да') then createRecord.Materialist\_chkbx.Checked := True

else createRecord.Materialist\_chkbx.Checked := False;

createRecord.First\_release.Text:= mRec.Cells[3, mRec.Row];

createRecord.Dir\_of\_philosophy\_label\_list.Text:= mRec.Cells[4, mRec.Row];

//устанавливаем ModalResult редактора в mrNone:

createRecord.ModalResult:= mrNone;

//теперь выводим форму:

createRecord.ShowModal;

//сохраняем в сетку возможные изменения,

//если пользователь нажал "Сохранить":

if createRecord.ModalResult = mrOk then begin

mRec.Cells[0, mRec.Row]:= createRecord.Name\_edit.Text;

mRec.Cells[1, mRec.Row]:= createRecord.Year\_of\_foundation\_edit.Text;

if (createRecord.Materialist\_chkbx.Checked) then mRec.Cells[2, mRec.RowCount-1]:= 'Да'

else mRec.Cells[2, mRec.RowCount-1]:='Нет';

mRec.Cells[3, mRec.Row]:= createRecord.First\_release.Text;

mRec.Cells[4, mRec.Row]:= createRecord.Dir\_of\_philosophy\_label\_list.Text;

end;

end;

procedure TmForm.PanelClick(Sender: TObject);

begin

end;

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

var

new\_Group: Group; //для очередной записи

f: file of Group; //файл данных

i: integer; //счетчик цикла

begin

//если строки данных пусты, просто выходим:

if mRec.RowCount = 1 then exit;

//иначе открываем файл для записи:

try

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

Rewrite(f);

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

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

//получаем данные текущей записи:

new\_Group.Name:= mRec.Cells[0, i];

new\_Group.Year\_of\_foundation:= StrToInt(mRec.Cells[1, i]);

if (mRec.Cells[2, mRec.RowCount-1] = 'Да') then new\_Group.Exist := True

else new\_Group.Exist := False;

new\_Group.First\_release:= StrToInt(mRec.Cells[3, i]);

new\_Group.Genre:= mRec.Cells[4, i];

//записываем их:

Write(f, new\_Group);

end;

finally

CloseFile(f);

end;

end;

end.

# **Результат выполнения программы**

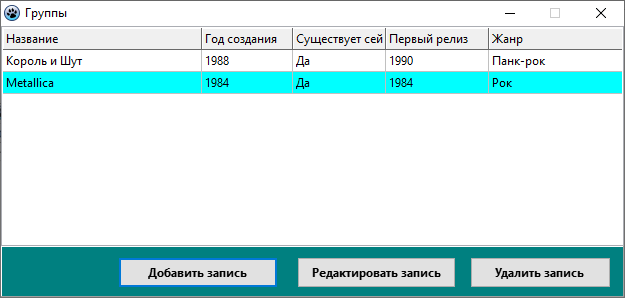


Рис. 1 – Результат выполнения программы

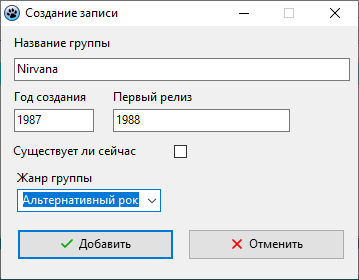


Рис. 2 – Результат выполнения программы

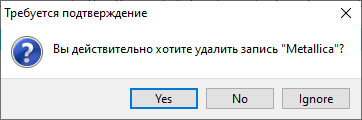


Рис. 3 – Результат выполнения программы

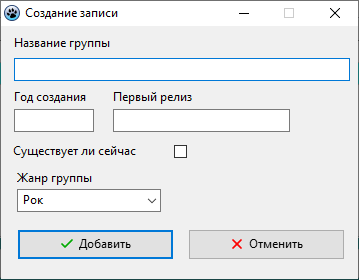


Рис. 4 – Результат выполнения программы

# **Вывод**

В процессе выполнения данной лабораторной работы была достигнута основная цель: получение навыков организации хранения и использования информации посредством типизированных файлов, а также закрепление навыков создания пользовательского интерфейса. Работа включала в себя разработку программы на языке Delphi в среде Lazarus, которая позволяла извлекать набор записей из типизированного файла, визуализировать данный набор, изменять данные и сохранять произведённые изменения. В записи содержалось не менее пяти элементов, среди которых было минимум три различных типа данных. Было получено представление об организации хранения и использования информации посредством типизированных файлов, закреплены навыки создания пользовательского интерфейса.