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»**

Выполнил: студент учебной группы

ИСПк-205-52-00

Сальченко Егор Вадимович

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2024

**Цель работы**

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

**Формулировка задания**

Задание:

* ﻿﻿﻿﻿﻿﻿Сформировать структуру записи, с которой будет осуществляться работа, и согласовать ее с преподавателем.
* ﻿Разработать приложение, позволяющее извлекать набор записей из типизированного файла, визуализировать данный набор, а также позволяющее изменять данных и выполнять сохранение проделанных изменений.

**Описание алгоритма и ответы на вопросы**

### 1. Запуск приложения и настройка интерфейса (FormCreate)

* **Что происходит**: При запуске программы настраивается таблица StringGrid1 для отображения данных о фильмах.
* **Детали**:
  + Устанавливается 5 колонок: "ID", "Название", "Режиссер", "Год выпуска", "Рейтинг".
  + Первая строка фиксируется как заголовок (FixedRows := 1).
  + Переменная CurrentRow инициализируется значением -1, что означает, что ни одна строка не выбрана.

### 2. Загрузка данных из файла (ButtonLoadClick)

* Старт: Программа запускается, показывает таблицу с колонками: ID, название, режиссер, год, рейтинг.
* Загрузка: Жмёшь кнопку — она берёт данные о фильмах из файла films.csv и загружает в таблицу. Если файла нет, скажет: "Где файл-то?".
* Сохранение: Нажал "Сохранить" — данные из таблицы записываются обратно в файл. Если в названии есть запятая, она ставит кавычки, чтобы всё было аккуратно.
* Редактирование: Выбираешь фильм в таблице, меняешь что надо в полях, жмёшь "Редактировать". Если что-то пустое — попросит заполнить.
* Выбор: Кликнул на строку в таблице — данные сразу в поля ввода подтягиваются, чтобы править было удобно.
* Очистка: Есть кнопка, чтобы поля ввода обнулить.

**Код программы**

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, Forms, Controls, Grids, StdCtrls, Dialogs;

type

// структура записи о фильме

TFilm = record

ID: Integer;

Title: string;

Director: string;

Year: Integer;

Rating: Double;

end;

{ TForm1 }

TForm1 = class(TForm)

ButtonEdit: TButton;

ButtonSave: TButton;

ButtonLoad: TButton;

EditRating: TEdit;

EditYear: TEdit;

EditDirector: TEdit;

EditTitle: TEdit;

GroupBox1: TGroupBox;

LabelRating: TLabel;

LabelYear: TLabel;

LabelDirector: TLabel;

LabelTitle: TLabel;

StringGrid1: TStringGrid;

procedure FormCreate(Sender: TObject);

procedure ButtonLoadClick(Sender: TObject);

procedure ButtonSaveClick(Sender: TObject);

procedure ButtonEditClick(Sender: TObject);

procedure GroupBox1Click(Sender: TObject);

procedure LabelTitleClick(Sender: TObject);

procedure StringGrid1SelectCell(Sender: TObject; ACol, ARow: Integer;

var CanSelect: Boolean);

private

Films: array of TFilm; // массив фильмов

CurrentRow: Integer; // выбранная строка

procedure ClearInputs; // очистка полей

function EscapeCSV(const S: string): string; // экранирование строк для CSV

public

end;

var

Form1: TForm1;

implementation

{$R \*.lfm}

{ TForm1 }

// настройка таблицы при запуске

procedure TForm1.FormCreate(Sender: TObject);

begin

StringGrid1.ColCount := 5;

StringGrid1.FixedRows := 1;

StringGrid1.Cells[0, 0] := 'ID';

StringGrid1.Cells[1, 0] := 'Название';

StringGrid1.Cells[2, 0] := 'Режиссер';

StringGrid1.Cells[3, 0] := 'Год выпуска';

StringGrid1.Cells[4, 0] := 'Рейтинг';

CurrentRow := -1;

end;

// экранирование строк для CSV

function TForm1.EscapeCSV(const S: string): string;

begin

Result := S;

if Pos(',', S) > 0 then

Result := '"' + StringReplace(S, '"', '""', [rfReplaceAll]) + '"';

end;

// загрузка данных из csv

procedure TForm1.ButtonLoadClick(Sender: TObject);

var

F: TextFile;

Line: string;

Parts: TStringList;

i: Integer;

begin

if not FileExists('films.csv') then

begin

ShowMessage('Файл films.csv не найден!');

Exit;

end;

AssignFile(F, 'films.csv');

Reset(F);

SetLength(Films, 0);

StringGrid1.RowCount := 1; // очищаем таблицу (оставляем только заголовки)

Parts := TStringList.Create;

Parts.StrictDelimiter := True; // важно!

Parts.Delimiter := ',';

try

while not EOF(F) do

begin

ReadLn(F, Line);

Parts.DelimitedText := Line;

if Parts.Count = 5 then

begin

// добавляем в массив

SetLength(Films, Length(Films) + 1);

with Films[High(Films)] do

begin

ID := StrToIntDef(Parts[0], 0);

Title := Parts[1];

Director := Parts[2];

Year := StrToIntDef(Parts[3], 0);

Rating := StrToFloatDef(Parts[4], 0.0);

end;

// добавляем в таблицу

StringGrid1.RowCount := StringGrid1.RowCount + 1;

StringGrid1.Cells[0, StringGrid1.RowCount - 1] := Parts[0];

StringGrid1.Cells[1, StringGrid1.RowCount - 1] := Parts[1];

StringGrid1.Cells[2, StringGrid1.RowCount - 1] := Parts[2];

StringGrid1.Cells[3, StringGrid1.RowCount - 1] := Parts[3];

StringGrid1.Cells[4, StringGrid1.RowCount - 1] := Parts[4];

end;

end;

finally

Parts.Free;

CloseFile(F);

end;

ShowMessage('Фильмы загружены!');

end;

// сохранение в csv

procedure TForm1.ButtonSaveClick(Sender: TObject);

var

F: TextFile;

i: Integer;

ratingStr: string;

begin

AssignFile(F, 'films.csv');

Rewrite(F);

for i := 0 to High(Films) do

begin

ratingStr := Format('%.1f', [Films[i].Rating]);

WriteLn(F, IntToStr(Films[i].ID) + ',' +

EscapeCSV(Films[i].Title) + ',' +

EscapeCSV(Films[i].Director) + ',' +

IntToStr(Films[i].Year) + ',' +

EscapeCSV(ratingStr));

end;

CloseFile(F);

ShowMessage('Данные сохранены в films.csv');

end;

// редактирование выбранной строки

procedure TForm1.ButtonEditClick(Sender: TObject);

begin

if CurrentRow < 0 then

begin

ShowMessage('Выберите запись для редактирования!');

Exit;

end;

if (EditTitle.Text = '') or (EditDirector.Text = '') or

(EditYear.Text = '') or (EditRating.Text = '') then

begin

ShowMessage('Заполните все поля!');

Exit;

end;

// обновляем данные в массиве

with Films[CurrentRow] do

begin

Title := EditTitle.Text;

Director := EditDirector.Text;

Year := StrToIntDef(EditYear.Text, 0);

Rating := StrToFloatDef(EditRating.Text, 0.0);

end;

// обновляем данные в таблице

StringGrid1.Cells[1, CurrentRow + 1] := Films[CurrentRow].Title;

StringGrid1.Cells[2, CurrentRow + 1] := Films[CurrentRow].Director;

StringGrid1.Cells[3, CurrentRow + 1] := IntToStr(Films[CurrentRow].Year);

StringGrid1.Cells[4, CurrentRow + 1] := FloatToStr(Films[CurrentRow].Rating);

ClearInputs;

ShowMessage('Запись обновлена');

end;

procedure TForm1.GroupBox1Click(Sender: TObject);

begin

end;

procedure TForm1.LabelTitleClick(Sender: TObject);

begin

end;

// при выборе строки — заполняем поля

procedure TForm1.StringGrid1SelectCell(Sender: TObject; ACol, ARow: Integer;

var CanSelect: Boolean);

begin

if ARow > 0 then

begin

if ARow - 1 <= High(Films) then

begin

CurrentRow := ARow - 1;

EditTitle.Text := Films[CurrentRow].Title;

EditDirector.Text := Films[CurrentRow].Director;

EditYear.Text := IntToStr(Films[CurrentRow].Year);

EditRating.Text := FloatToStr(Films[CurrentRow].Rating);

end;

end;

end;

// очистка полей

procedure TForm1.ClearInputs;

begin

EditTitle.Text := '';

EditDirector.Text := '';

EditYear.Text := '';

EditRating.Text := '';

CurrentRow := -1;

end;

end.

**Результат выполнения программы**

