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**

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

Выполнил: студент учебной группы

ИСПк-202-52-00

Колобов Александр Алексеевич

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:**

1. Сформировать структуру записи, с которой будет осуществляться работа, и согласовать ее с преподавателем.

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

Код программы

**unit MainUnit;**

**{$mode objfpc}{$H+}**

**interface**

**uses**

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

**Grids, Edit;**

**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);**

**procedure Panel1Click(Sender: TObject);**

**private**

**public**

**end;**

**type**

**Books = record**

**Name: string[40];**

**Author: string[40];**

**Note: string[10];**

**Year: integer;**

**Cost: real;**

**end; //record**

**var**

**fMain: TfMain;**

**adres: string; //адрес, откуда запущена программа**

**implementation**

**{$R \*.lfm}**

**{ TfMain }**

**procedure TfMain.bAddClick(Sender: TObject);**

**begin**

**//очищаем поля, если там что-то есть:**

**fEdit.book\_name.Text:= '';**

**fEdit.book\_author.Text:= '';**

**//устанавливаем ModalResult редактора в mrNone:**

**fEdit.ModalResult:= mrNone;**

**//теперь выводим форму:**

**fEdit.ShowModal;**

**//если пользователь ничего не ввел - выходим:**

**if (fEdit.book\_author.Text= '') or (fEdit.book\_author.Text= '') then exit;**

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

**if fEdit.ModalResult <> mrOk then exit;**

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

**SG.RowCount:= SG.RowCount + 1;**

**SG.Cells[0, SG.RowCount-1]:= fEdit.book\_name.Text;**

**SG.Cells[1, SG.RowCount-1]:= fEdit.book\_author.Text;**

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

**SG.Cells[3, SG.RowCount-1]:= fEdit.book\_year.Text;**

**SG.Cells[4, SG.RowCount-1]:= fEdit.book\_cost.Text;**

**end;**

**procedure TfMain.bDelClick(Sender: TObject);**

**begin**

**begin**

**//если данных нет - выходим:**

**if SG.RowCount = 1 then exit;**

**//иначе выводим запрос на подтверждение:**

**if MessageDlg('Требуется подтверждение',**

**'Вы действительно хотите удалить контакт "' +**

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

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

**SG.DeleteRow(SG.Row);**

**end;**

**end;**

**procedure TfMain.bEditClick(Sender: TObject);**

**begin**

**//если данных в сетке нет - просто выходим:**

**if SG.RowCount = 1 then exit;**

**//иначе записываем данные в форму редактора:**

**fEdit.book\_name.Text:= SG.Cells[0, SG.Row];**

**fEdit.book\_author.Text:= SG.Cells[1, SG.Row];**

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

**fEdit.book\_year.Text:= SG.Cells[3, SG.Row];**

**fEdit.book\_cost.Text:= SG.Cells[4, SG.Row];**

**//устанавливаем ModalResult редактора в mrNone:**

**fEdit.ModalResult:= mrNone;**

**//теперь выводим форму:**

**fEdit.ShowModal;**

**//сохраняем в сетку возможные изменения,**

**//если пользователь нажал "Сохранить":**

**if fEdit.ModalResult = mrOk then begin**

**SG.Cells[0, SG.Row]:= fEdit.book\_name.Text;**

**SG.Cells[1, SG.Row]:= fEdit.book\_author.Text;**

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

**SG.Cells[3, SG.Row]:= fEdit.book\_year.Text;**

**SG.Cells[4, SG.Row]:= fEdit.book\_cost.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**

**MyBooks: Books; //для очередной записи**

**f: file of Books; //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 + 'books.dat');**

**Rewrite(f);**

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

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

**//получаем данные текущей записи:**

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

**MyBooks.Author := SG.Cells[1, i];**

**MyBooks.Note:= SG.Cells[2, i];**

**MyBooks.Year:= strtoint(SG.Cells[3, i]);**

**MyBooks.Cost:= strtofloat(SG.Cells[4, i]);**

**//записываем их:**

**Write(f, MyBooks);**

**end;**

**finally**

**CloseFile(f);**

**end;**

**end;**

**procedure TfMain.FormCreate(Sender: TObject);**

**var**

**MyBooks: Books; //для очередной записи**

**f: file of Books; //файл данных**

**i: integer; //счетчик цикла**

**begin**

**//сначала получим адрес программы:**

**adres:= ExtractFilePath(ParamStr(0));**

**//настроим сетку:**

**SG.Cells[0, 0]:= 'название';**

**SG.Cells[1, 0]:= 'автор';**

**SG.Cells[2, 0]:= 'наличие';**

**SG.Cells[3, 0]:= 'год выпуска';**

**SG.Cells[4, 0]:= 'цена';**

**SG.ColWidths[0]:= 250;**

**SG.ColWidths[1]:= 250;**

**SG.ColWidths[2]:= 100;**

**SG.ColWidths[3]:= 100;**

**SG.ColWidths[4]:= 100;**

**//если файла данных нет, просто выходим:**

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

**//иначе файл есть, открываем его для чтения и**

**//считываем данные в сетку:**

**try**

**AssignFile(f, adres + 'books.dat');**

**Reset(f);**

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

**while not Eof(f) do begin**

**//считываем новую запись:**

**Read(f, MyBooks);**

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

**SG.RowCount:= SG.RowCount + 1;**

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

**SG.Cells[1, SG.RowCount-1]:= MyBooks.Author;**

**SG.Cells[2, SG.RowCount-1]:= MyBooks.Note;**

**SG.Cells[3, SG.RowCount-1]:= inttostr(MyBooks.Year);**

**SG.Cells[4, SG.RowCount-1]:= floattostr(MyBooks.Cost);**

**end;**

**finally**

**CloseFile(f);**

**end;**

**end;**

**procedure TfMain.Panel1Click(Sender: TObject);**

**begin**

**end;**

**end.**

Результат выполнения программы

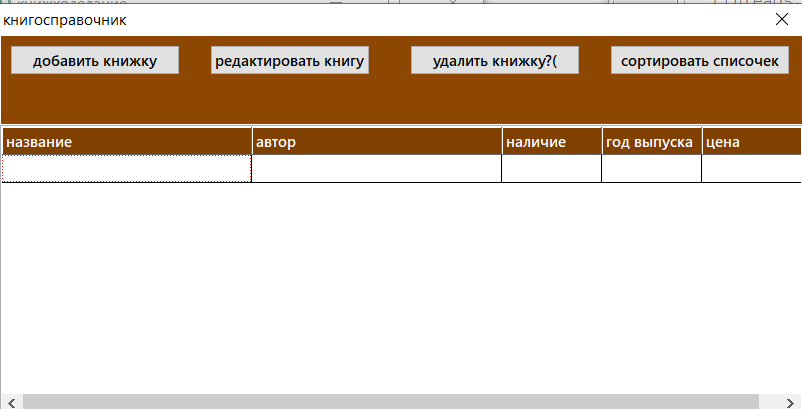


Рисунок 1 - Результат выполнения программы

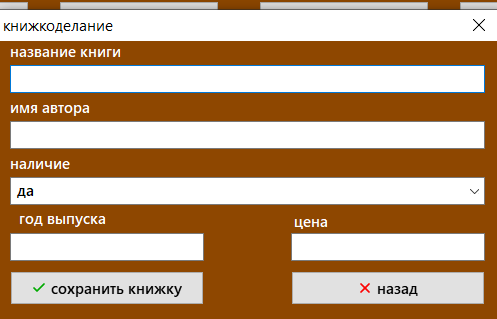


Рисунок 2 – Результат выполнения программы

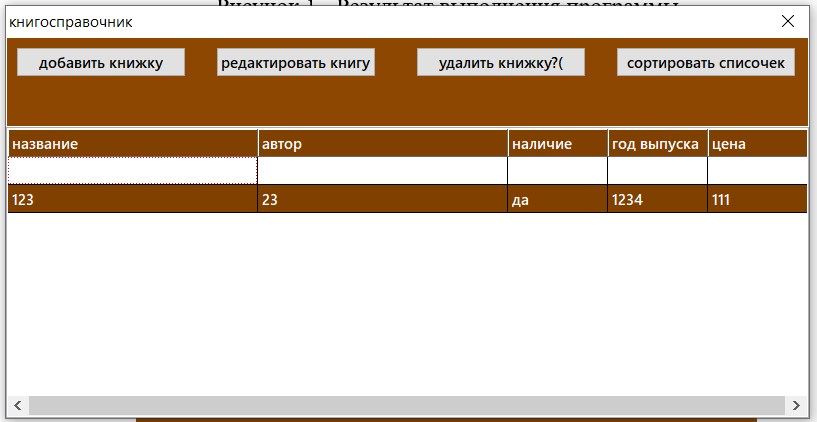


Рисунок 3 – Результат выполнения программы

Вывод

В ходе выполнения данной работы мы смогли применить изученный материал лабораторных занятий на практике, процедуры, условный оператор, цикл while, repeat, for, создание переменных и операторов ввода и вывода, а также мы построили схему алгоритма с помощью бесплатного онлайн-сервиса app.diagrams.net.

В результате выполнения данной работы получили базовые сведения о наиболее известных алгоритмах сортировки, изучили принципы работы с текстовыми файлами, а также выполнили данное нам практическое задание, использовав изученный материал на практике.

Таким образом, выполнение домашней контрольной работы №7 принесло нам ценный опыт и знания, которые будут полезны в нашем дальнейшем образовании. Мы научились применять определённые базовые конструкции языка программирования, а также разрабатывать схему алгоритма для решения задачи. Этот опыт будет полезен нам будущем для решений новых задач и работой над реальными проектами.