Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №8**

**«ОСНОВЫ СОБЫТИЙНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ»**

**ПО МДК 05.02 РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ**

Выполнил: студент учебной группы

ИСПк-101-51-00

Костин Борис Сергеевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

**Цель работы:**

Получение базовых навыков реализации приложения с графическим интерфейсом пользователя на основе событийно-ориентированной парадигмы.

**Задания:**

Вариант 15

1.Реализовать приложение “Калькулятор простых и сложных процентов”

2.За период начисления процентов должен приниматься год.

3.Предусмотреть возможность вывода дополнительной информации по расчетам.

Описание алгоритма

В простых процентах – берется начальная сумма в рублях, к ней прибавляется деление на сотню скобок с перемножением начальной суммы, годовой процентной ставки и времени в годах. Стабильный постоянный одинаковый рост.

В сложных процентах – берется начальная сумма, она умножается на скобки в степени равной количеству месяцев, в самих скобках к единице прибавляется деление на 12 деления годовой процентной ставки на 100. Частично стабильный увеличивающийся по экспоненте рост.

**Код программы:**

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

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

Buttons, Math;

type

{ TForm1 }

TForm1 = class(TForm)

btnClear: TButton;

btnClose: TBitBtn;

btnCalculate: TButton;

cbxCalculationType: TComboBox;

edtPrincipal: TEdit;

edtRate: TEdit;

edtTime: TEdit;

LabelPrincipal: TLabel;

LabelProcent: TLabel;

LabelRate: TLabel;

LabelTime: TLabel;

memResult: TMemo;

procedure btnCalculateClick(Sender: TObject);

procedure btnClearClick(Sender: TObject);

procedure btnCloseClick(Sender: TObject);

procedure edtPrincipalKeyPress(Sender: TObject; var Key: char);

procedure edtRateKeyPress(Sender: TObject; var Key: char);

procedure edtTimeKeyPress(Sender: TObject; var Key: char);

procedure FormCreate(Sender: TObject);

private

public

end;

var

Form1: TForm1;

implementation

{$R \*.lfm}

{ TForm1 }

procedure TForm1.btnClearClick(Sender: TObject);

begin

edtPrincipal.Clear;

edtRate.Clear;

edtTime.Clear;

memResult.Clear

end;

procedure TForm1.btnCalculateClick(Sender: TObject);

const

MaxResult = 1.1e38; // Maximum representable positive finite value for a Double data type in Pascal

MinResult = -1.1e38;

var

Principal, Rate, Time, Result: Double;

begin

if (edtPrincipal.Text = '') or (edtRate.Text = '') or (edtTime.Text = '') then

begin

ShowMessage('Нужно заполнить все поля');

end

else

begin

try

Principal := StrToFloat(edtPrincipal.Text);

Rate := StrToFloat(edtRate.Text);

Time := StrToFloat(edtTime.Text);

case cbxCalculationType.ItemIndex of

0: // Простые проценты

begin

Result := Principal + (Principal \* Time \* Rate )/100;

memResult.Lines.Clear;

if (Result <= MaxResult) and (Result >= MinResult) then

memResult.Lines.Add('Результат: ' + FormatFloat('0.00', Result))

else

ShowMessage('Переполнение числа');

end;

1: // Сложные проценты

begin

Result := Principal \* Power((1+((Rate/100)/12)),Time\*12);

memResult.Lines.Clear;

if (Result <= MaxResult) and (Result >= MinResult) then

memResult.Lines.Add('Результат: ' + FormatFloat('0.00', Result))

else

ShowMessage('Переполнение числа');

end;

end;

except

on E: Exception do

ShowMessage('Ошибка: ' + E.Message);

end;

end;

end;

procedure TForm1.btnCloseClick(Sender: TObject);

begin

close;

end;

procedure TForm1.edtPrincipalKeyPress(Sender: TObject; var Key: char);

begin

if not (Key in ['0'..'9', ',', '.', #8, #13, #27]) then

Key := #0;

end;

procedure TForm1.edtRateKeyPress(Sender: TObject; var Key: char);

begin

if not (Key in ['0'..'9', ',', '.', #8, #13, #27]) then

Key := #0;

end;

procedure TForm1.edtTimeKeyPress(Sender: TObject; var Key: char);

begin

if not (Key in ['0'..'9', ',', '.', #8, #13, #27]) then

Key := #0;

end;

procedure TForm1.FormCreate(Sender: TObject);

begin

cbxCalculationType.Items.Add('Простые проценты');

cbxCalculationType.Items.Add('Сложные проценты');

cbxCalculationType.ItemIndex := 0;

end;

end.

**Результат выполнения программы:**

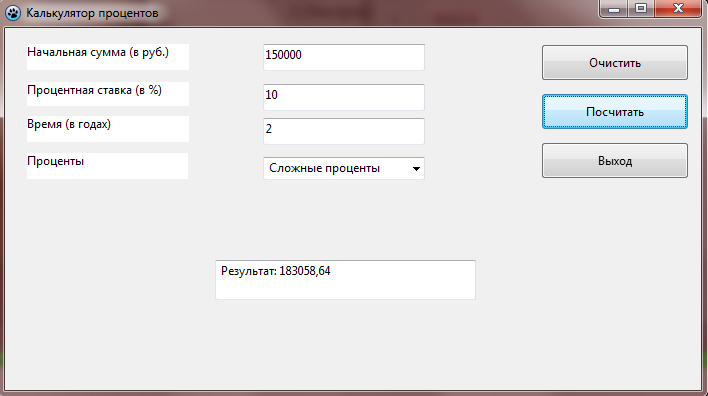


Рисунок 1 – Результат выполнения программы

**Вывод:**

В ходе выполнения домашней контрольной работы №8 была более подробно изучена IDE Lazarus. В первом этапе работы был выполнен анализ предметной области: формулы, компоненты. Во втором этапе составлен интерфейс: добавлены кнопки, поля, группы, лейблы. Третий этап заключался в написании логики и масштабирования программы. Трудностей в работе не возникало. Таким образом, в домашней контрольной работе №8 поставленные задания были выполнены мной в полном объеме.