Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №7**

**«ИСЛЕДОВАНИЕ ФРАКТАЛА»**

**ПО МДК 05.02 РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ**

Выполнил: студент учебной группы

ИСПк-204-52-00

Воробьев Алексей Владимирович

Преподаватель:

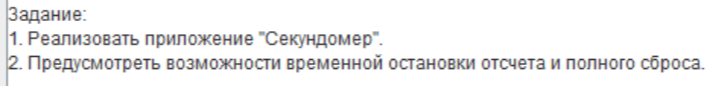
Сергеева Елизавета Григорьевна

Киров

2023

1. Цель работы

получение базовых навыков реализации приложений с графическим интерфейсом пользователя на основе событийно-ориентированной парадигмы.



1. Описание алгоритма

В коде описаны 5 элементов:TButton,TLabel,TMemo,TImage,TTimer. Есть 3 кнопки, сброс очищает TMemo, стоп останавливает TLabel который обновляется TTimer и заносит результат в TMemo, старт который возобновляет Tlable.

1. Код программы

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

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

type

{ TSecond }

TSecond = class(TForm)

btnStart: TButton;

btnStop: TButton;

btnReset: TButton;

Image: TImage;

lblTimer: TLabel;

memoLaps: TMemo;

TimerTime: TTimer;

procedure btnResetClick(Sender: TObject);

procedure btnStartClick(Sender: TObject);

procedure btnStopClick(Sender: TObject);

procedure TimerTimeTimer(Sender: TObject);

private

{ private declarations }

FStartTime: TDateTime;

FLapStartTime: TDateTime;

FTotalElapsed: TDateTime;

elapsedTime: string;

public

{ public declarations }

end;

var

Second: TSecond;

implementation

{$R \*.lfm}

{ TSecond }

procedure TSecond.btnStartClick(Sender: TObject);

begin

FStartTime := Now;

FLapStartTime := FStartTime;

FTotalElapsed := 0;

TimerTime.Enabled := True;

btnStart.Enabled := False;

btnStop.Enabled := True;

btnReset.Enabled := True;

end;

procedure TSecond.btnStopClick(Sender: TObject);

begin

TimerTime.Enabled := False;

btnStart.Enabled := True;

btnStop.Enabled := False;

memoLaps.Lines.Add(lblTimer.Caption);

end;

procedure TSecond.btnResetClick(Sender: TObject);

begin

TimerTime.Enabled := False;

FStartTime := 0;

FLapStartTime := 0;

FTotalElapsed := 0;

lblTimer.Caption := '00:00:00.000';

memoLaps.Clear;

btnStart.Enabled := True;

btnStop.Enabled := False;

btnReset.Enabled := False;

end;

procedure TSecond.TimerTimeTimer(Sender: TObject);

var

elapsed: TDateTime;

lapElapsed: TDateTime;

begin

if TimerTime.Enabled then

begin

elapsed := Now - FStartTime;

lapElapsed := Now - FLapStartTime;

FTotalElapsed := FTotalElapsed + lapElapsed;

lblTimer.Caption := FormatDateTime('hh:nn:ss.zzz', FTotalElapsed);

FLapStartTime := Now;

end;

end;

end.

5. Результат выполнения программы

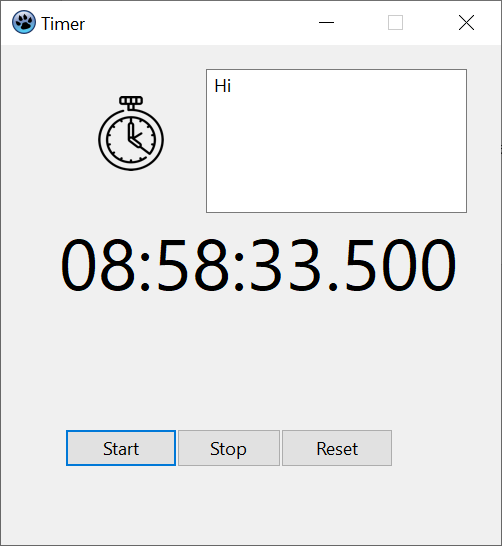


Рисунок 1 – Результаты выполнение программы

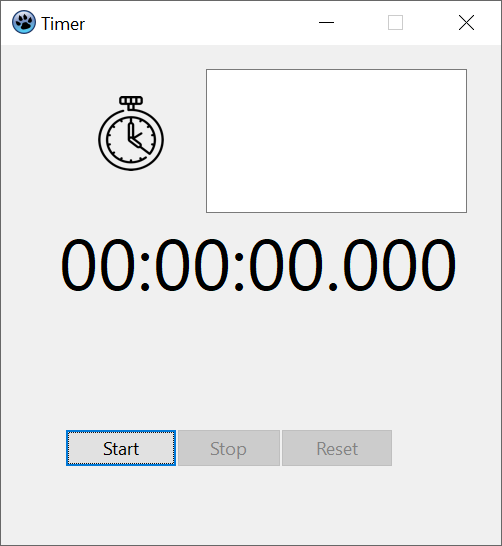


Рисунок 2 – Результаты выполнение программы

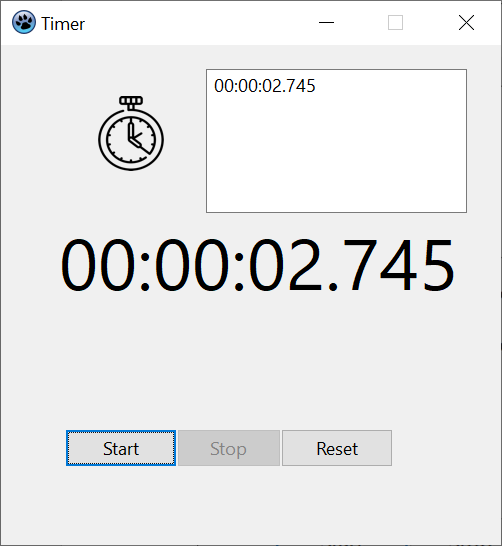


Рисунок 3 – Результаты выполнение программы

6. Вывод

В этой домашней контрольной работе использовался IDE Lazarus.

Программа начинается с того что в TLable ставится текущее время, а в TMemo выводится Hi. С помощью кнопок осуществляется управление таймером, старт начинает обновление таймера, стоп останавливает обновление таймера, сброс стирает данные из TMemo.

В ходе разработки кода встретились проблемы с подвязыванием TTimer к TLable.