Лабораторная работа №6

Менеджер задач

unit Unit1;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,

Dialogs, XPMan, StdCtrls, jpeg, ExtCtrls, ComCtrls,ShellApi;

type

TForm1 = class(TForm)

Timer1: TTimer;

Timer3: TTimer;

Timer4: TTimer;

Panel2: TPanel;

Panel4: TPanel;

Button1: TButton;

Button2: TButton;

GroupBox1: TGroupBox;

Label1: TLabel;

Label2: TLabel;

Edit1: TEdit;

Edit2: TEdit;

Edit3: TEdit;

DateTimePicker1: TDateTimePicker;

DateTimePicker2: TDateTimePicker;

DateTimePicker3: TDateTimePicker;

БУДИЛЬНИК: TLabel;

DateTimePicker4: TDateTimePicker;

Label3: TLabel;

Edit4: TEdit;

Button3: TButton;

Label4: TLabel;

Label5: TLabel;

Edit5: TEdit;

Edit6: TEdit;

UpDown1: TUpDown;

UpDown2: TUpDown;

Panel1: TPanel;

Label6: TLabel;

CheckBox1: TCheckBox;

CheckBox2: TCheckBox;

DateTimePicker5: TDateTimePicker;

OpenComposition: TOpenDialog;

Label8: TLabel;

Timer2: TTimer;

Label7: TLabel;

procedure Timer1Timer(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure Timer2Timer(Sender: TObject);

procedure Button3Click(Sender: TObject);

procedure Timer4Timer(Sender: TObject);

procedure openprog(i:string);

procedure Button1Click(Sender: TObject);

procedure Button4Click(Sender: TObject);

procedure Timer3Timer(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form1: TForm1;

min: integer; // минут

sec: integer; // секунд

implementation

{$R \*.dfm}

procedure ShowTime;

var

buf: string[20];

begin

buf := 'Таймер ';

// минуты и секунды выводим двумя цифрами

if min < 10 then

buf := buf + '0' + IntToStr(min) + ' : '

else

buf := buf + IntToStr(min) + ' : ';

if sec < 10 then

buf := buf + '0' + IntToStr(sec)

else

buf := buf + IntToStr(sec);

Form1.Caption := buf;

end;

procedure TForm1.Button1Click(Sender: TObject);

begin

close;

end;

procedure TForm1.Button2Click(Sender: TObject);

begin

if Form1.Timer1.Enabled then

// таймер работает, надо остановить

begin

Timer1.Enabled := False; // стоп

Form1.Caption := 'Таймер';

Button2.Caption := 'Пуск';

// разрешить ввод интервала

UpDown1.Enabled:= True;

UpDown2.Enabled:= True;

Edit5.Enabled := True;

Edit6.Enabled := True;

end

else

// таймер стоит, надо запустить

begin

min := UpDown1.Position;

sec := UpDown2.Position;

if (sec = 0) and (min = 0) then

begin

ShowMessage('Надо задать интервал!');

exit;

end;

Edit5.Enabled := False;

Edit6.Enabled := False;

UpDown1.Enabled:= False;

UpDown2.Enabled:= False;

Button2.Caption := 'Стоп';

Timer1.Enabled := True; // пуск таймера

ShowTime;

end;

end;

procedure TForm1.Button3Click(Sender: TObject);

begin

if OpenComposition.Execute then Edit4.Text:=OpenComposition.FileName;

end;

procedure TForm1.Button4Click(Sender: TObject);

begin

if(label8.caption)=timetostr(DateTimePicker4.Time) then

ShellExecute(handle, 'open', pchar(Edit4.Text), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm1.Timer1Timer(Sender: TObject);

begin

sec := sec - 1;

if (min = 0) and (sec = 0) then

// заданный интервал истек

begin

Timer1.Enabled := False; // остановить таймер

UpDown1.Enabled := True;

UpDown2.Enabled := True;

Edit5.Enabled := True;

Edit6.Enabled := True;

Button2.Caption := 'Пуск';

ShowMessage('Заданный интервал истек!');

exit;

if (sec = 0) and (min > 0) then

begin

sec := 60;

min := min - 1;

end;

ShowTime; // показать, сколько времени осталось

end;

end;

procedure TForm1.openprog(i:string);

begin

ShellExecute(TForm(Owner).Handle,nil,PCHAR(i),nil,nil,SW\_SHOWNORMAL);

end;

procedure TForm1.Timer2Timer(Sender: TObject);

begin

label8.Caption:=Timetostr(time);

if(label8.caption)=timetostr(DateTimePicker1.Time) then

openprog(Edit1.text);

if(label8.caption)=timetostr(DateTimePicker2.Time) then

openprog(Edit2.text);

if(label8.caption)=timetostr(DateTimePicker3.Time) then

openprog(Edit3.text);

end;

procedure TForm1.Timer3Timer(Sender: TObject);

begin

if(label8.caption)=timetostr(DateTimePicker4.Time) then

ShellExecute(handle, 'open', pchar(Edit4.Text), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm1.Timer4Timer(Sender: TObject);

var

hToken: THandle;

tkp: TTokenPrivileges;

ReturnLength: Cardinal;

begin

if OpenProcessToken(GetCurrentProcess(), TOKEN\_ADJUST\_PRIVILEGES or

TOKEN\_QUERY, hToken) then

begin

LookupPrivilegeValue(nil, 'SeShutdownPrivilege', tkp.Privileges[0].Luid);

tkp.PrivilegeCount := 1;

tkp.Privileges[0].Attributes := SE\_PRIVILEGE\_ENABLED;

if AdjustTokenPrivileges(hToken, False, tkp, 0, nil, ReturnLength)

then if(label8.caption)=timetostr(DateTimePicker5.Time)

then if CheckBox1.Checked then ExitWindowsEx(EWX\_ShutDown or EWX\_FORCE,0)

else if CheckBox2.Checked then ExitWindowsEx(EWX\_REBOOT or EWX\_FORCE,0);

end;

end;

end.

