



Nazwa implementacji: Zegar wskazówkowy

Autor: Stanisław Ubermanowicz

Opis implementacji: Realizacja zegara wskazówkowego, startowanego bieżącym czasem systemowym. Zegar powinien mieć trzy wskazówki: godzinową, minutową i sekundową. Wszystkie elementy zegara powinny być wygenerowane z poziomu kodu źródłowego, jako właściwości obiektów Canvas.

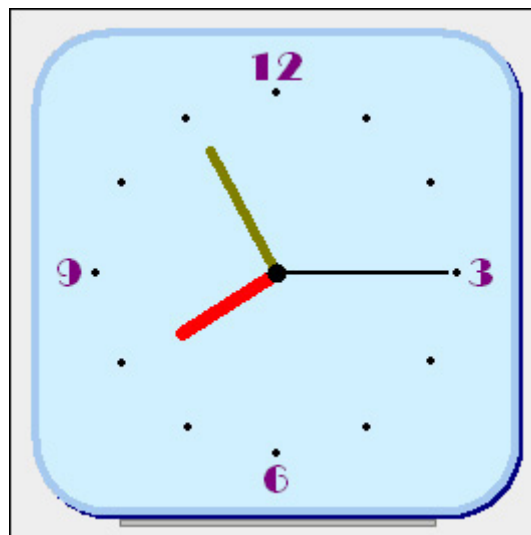
Przygotuj Okno o wymiarach 260x260 pikseli. Ustaw właściwość Position: poScreenCenter

Wprowadzaj lub uzupełniaj kod, starając się zrozumieć komentarze kodu oraz sposób definiowania właściwości obiektów ekranowych.

UWAGA! Po wpisaniu kodu źródłowego wprowadź powiązania obiektów ze zdarzeniami:

1. W polu Inspektora obiektów wskaź widżet Timer1, w zakładce Zdarzenia przy zdarzeniu OnTimer wybierz z listy wywołanie procedury Timer1Timer. Ma to na celu uaktualnianie czasu co 1 sekundę.
2. W polu Inspektora obiektów wskaź okno Form1, w zakładce Zdarzenia przy zdarzeniu OnPaint wybierz z listy wołanie procedury Timer1Timer. Ma to na celu wymuszenie odmalowywania obiektów przysłoniętych.

Wygląd przykładowego zegara po uruchomieniu



Kod źródłowy w środowisku Lazarus & FreePascal

```
unit zegarw1; { autor implementacji: Stanisław Ubermanowicz }
{$mode objfpc}{$H+}
interface
uses
  SysUtils, Forms, Graphics, ExtCtrls, DateUtils, Classes; { dodano DateUtils }
type { Definicja Okna i Timera oraz procedury obsługi Timera }
  TForm1 = class(TForm)
    Timer1: TTimer;
    procedure Timer1Timer(Sender: TObject);
  end;
var { Definicja zmiennych i ich typów }
  Form1: TForm1;
  Czas: TDateTime; { Zmienna do której pobierany jest czas systemowy }
1
```





```

pSek: Integer; { Pozycja wskazówki sekundowej, numerowana od 0 do 59 }
pMin: Integer; { Pozycja wskazówki minutowej, numerowana od 0 do 719 }
pGodz: Integer; { Pozycja wskazówki godzinowej, numerowana od 0 do 719 }
x, y: Integer; { Zmienne pomocnicze do obliczania położenia obiektów }
i: Integer; { Zmienna pomocnicza iteracji }
implementation
procedure TForm1.Timer1Timer(Sender: TObject); { Obsługa Timera co 1 sekundę }
begin
  Canvas.Pen.Color:= clNavy; Canvas.Pen.Width:= 4; { Kolor i grubość konturu }
  Canvas.RoundRect(12,12,252,252,72,72); { Rysuj figurę służącą tylko jako cień }
  Canvas.Pen.Color:= clSkyBlue; { Kolor konturu tarczy zegara }
  Canvas.Brush.Color:= $00FFF0D0; { Kolor wypełnienia tarczy zegara }
  Canvas.RoundRect(10,10,250,250,72,72); { Rysuj figurę imitującą tarczę zegara }
  Canvas.Font.Color:= clPurple; Canvas.Font.Size:= 16; { Kolor i rozmiar czcionek }
  Canvas.Font.Name:= 'Broadway'; { Nazwa czcionki wykorzystanej do zegara }
  Canvas.TextOut(116,15,'12'); { Wpisz na tarczę zegara liczby: 12, 3, 6 i 9 }
  Canvas.TextOut(226,118,'3'); { współrzędne tekstu zależą od użytych czcionek }
  Canvas.TextOut(123,221,'6'); Canvas.TextOut(20,118,'9');
  Canvas.Pen.Width:= 1; { Ustaw minimalną grubość konturu obiektów }
  Canvas.Pen.Color:= clGray; Canvas.Brush.Color:= clSilver; { Kolory podstawy zegara }
  Canvas.Rectangle(52,253,210,257); { Rysuj prostokąt imitujący podstawę zegara }
  Canvas.Pen.Color:= clBlack; Canvas.Brush.Color:= clBlack; { Ustaw kolor czarny }
  for i := 1 to 12 do { Pętla do rysowania 12 kropek na tarczy zegara }
  begin
    x:= 128+trunc(90*Sin(i*Pi/6)); { Oblicz pozycję kropki w poziomie }
    y:= 128-trunc(90*Cos(i*Pi/6)); { Oblicz pozycję kropki w pionie }
    Canvas.Ellipse(x,y,x+4,y+4); { Rysuj kropki o grubości 4 pikseli }
  end;
  Czas:= Now(); { Pobranie czasu systemowego do zmiennej wskazującej w danej chwili
  stan godzin, minut i sekund ze zmiennej określającej datę i czas }
  pSek:= SecondOf(Czas); { Pobranie stanu sekund i pozycji wskazówki sekundowej }
  pMin:= trunc(12*(MinuteOf(Czas) + pSek/60)); { Oblicz pozycję wskazówki minutowej }
  pGodz:= trunc(60*(HourOf(Czas) Mod 12 + pMin/720)); { Oblicz pozycję wskazówki godz }
  x:= 130+trunc(85*Sin(pSek*Pi/30)); { Pozycja od środka 130, długość wskazówki 85 }
  y:= 130-trunc(85*Cos(pSek*Pi/30)); { współrzędne XY z obrotu o kąt 6° * Pi/180° }
  Canvas.Pen.Color:= clBlack; Canvas.Pen.Width:= 2; { Kolor i grubość wskazówki }
  Canvas.Line(130,130,x,y); { Rysuj wskazówkę sekundową }
  x:= 130+trunc(70*Sin(pMin*Pi/360)); { Pozycja od środka 130, długość wskazówki 70 }
  y:= 130-trunc(70*Cos(pMin*Pi/360)); { współrzędne XY z obrotu o kąt 0.5° * Pi/180° }
  Canvas.Pen.Color:= clOlive; Canvas.Pen.Width:= 5; { Kolor i grubość wskazówki }
  Canvas.Line(130,130,x,y); { Rysuj wskazówkę minutową }
  x:= 130+trunc(56*Sin(pGodz*Pi/360)); { Pozycja od środka 130, długość wskazówki 56 }
  y:= 130-trunc(56*Cos(pGodz*Pi/360)); { współrzędne XY z obrotu o kąt 0.5° * Pi/180° }
  Canvas.Pen.Color:= clRed; Canvas.Pen.Width:= 7; { Kolor i grubość wskazówki }
  Canvas.Line(130,130,x,y); { Rysuj wskazówkę godzinową }
  Canvas.Pen.Color:= clBlack; Canvas.Pen.Width:= 2; { Kolor i grubość osi zegara }
  Canvas.Ellipse(126,126,135,135); { Rysuj oś wskazówek zegara }
end;
{$R *.lfm}
end.

```

