

Cvičení 7: Delphi – objekty `CheckBox`, `RadioButton`, `EditBox`

1 Opakování kreslení do `Image`

1. `Canvas` = plátno

Properties - Color, Caption

Nastavení typu čáry pro kreslení perem **Pen**:

- **barva** `Image1.Canvas.Pen.Color := 80;`
- **šířka** `Image1.Canvas.Pen.Width := 2;`
- **styl** `Image1.Canvas.Pen.Style := psDot;` další jsou `psDash`, `psSolid`

2. Základní útvary

(a) úsečka

- `Image1.Canvas.MoveTo(a,b);` ... počáteční bod
- `Image1.Canvas.LineTo(c,d);` ... koncový bod

(b) Obdélník

- `Image1.Canvas.Rectangle(a, b, c, d);`

(c) Elipsa

- `Image1.Canvas.Ellipse(a,b, c, d);` ... elipsa vepsaná obdélníku s rohy a, b a c, d.

(d) Lomená čára

- `Image1.Canvas.Polyline([Point(a,b), Point(c,d), Point(e, f)]);`

3. Propojení buttonu s procedurou

Button

- umístit na obrazovku
- popsat jej *Nakresli caru*
- **Events - OnClick** - napište název procedury *KresliCaru* a stiskněte *Enter* a mezi `begin` a `end` dopište proceduru.

2 Samostatný úkol

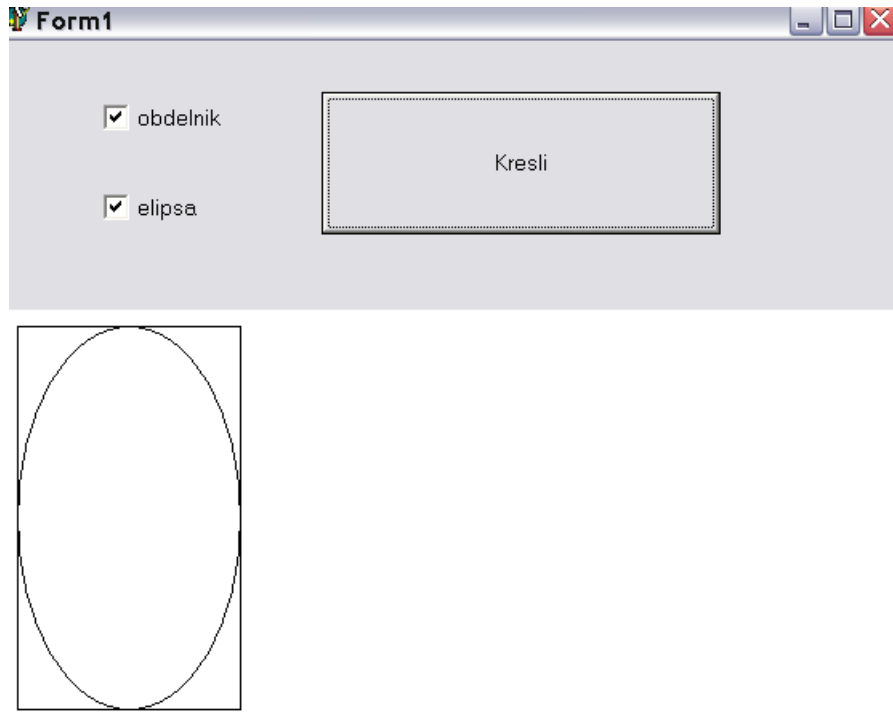
Vytvořte aplikaci obsahující buttony kreslí funkci, smaž a konec. Po stisknutí buttonu kreslí funkci se vykreslí graf funkce $y = |x|$, pro $x \in \langle -100, 100 \rangle$.

3 Objekt CheckBox - zaškrťovací políčko

- v záložce Standard
- Object Inspector → Text → popis u políčka
- při práci je automaticky pojmenováno CheckBox1 - nabývá hodnot true, false

Příklad:

Program, který po zaškrtnutí příslušného políčka vykreslí elipsu nebo obdélník.



1. vytvoříme dva CheckBoxy, popíšeme je
2. vytvoříme Image na spodní část formuláře, do kterého budeme kreslit
3. vytvoříme button s popiskou kresli a propojíme ho s procedurou kresli na událost OnClick

```
procedure TForm1.kresli(Sender: TObject);  
begin
```

```
if checkbox1.checked=true then  
image1.canvas.rectangle(10,10,150,250);
```

```
if checkbox2.checked=true then  
image1.canvas.Ellipse(10,10,150,250);
```

```
end;
```

4 Objekty RadioButton, RadioGroup

- v záložce Standard
- Object Inspector → Text → popis u políčka
- RadioButton nabývá hodnot true, false

```
RadioButton1.checked = true;  
RadioButton1.checked = false;
```

- RadioGroup slouží k zadávání více položek RadioButton pomocí volby Item v Object Inspector. Pokud není žádná z položek vybraná, nabývá vlastnost RadioGroup1.ItemIndex hodnoty -1. Pro jednotlivé položky pak hodnoty od jedné do počtu položek.

Příklad 1(obr. 1):

Udělejte program pomocí RadioGroup. Vypíše do zprávy, zda byla odpověď správná. Výpis zprávy provádí příkaz `showmessage('Toto se vypise');`

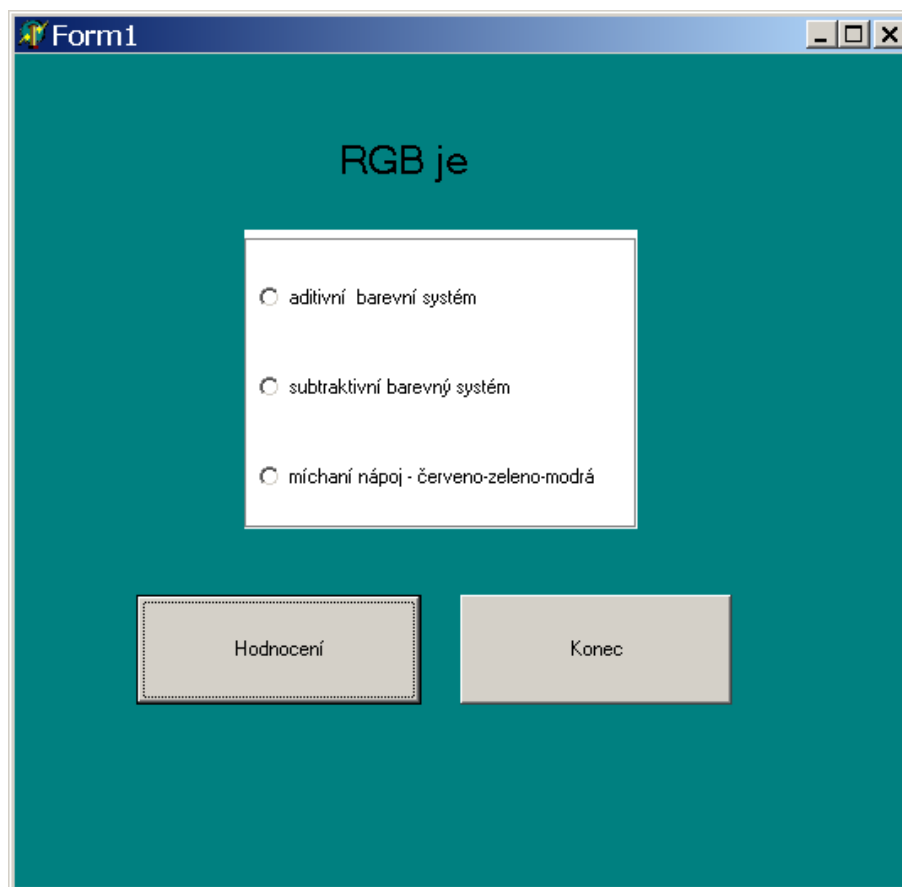
```
case RadioGroup1.ItemIndex of  
-1: showmessage('není vybrána žádná volba');  
0: showmessage('to je spravne');  
1: showmessage('skoro');  
2: showmessage('vtipalku :-');  
end;
```

Příklad 2:

Naprogramujte jednoduchý test, který vykreslí různé obrázky podle různých označení políček. Ošetřete, aby záraz nemohlo být zaškrtnuto více políček.

1. vytvoříme tři RadioButtony, popíšeme je v Object Inspectoru v položce text
2. ošetříme zaškrťování pouze jednoho políčka, v Object Inspectoru v Events na událost OnClick nastavíme proceduru `RadioButton1True` pro `RadioButton1`

```
procedure TForm1.RadioButton1True(Sender: TObject);  
begin  
  RadioButton1.checked:=true;  
  RadioButton2.checked:=false;  
  RadioButton3.checked:=false;  
end;
```

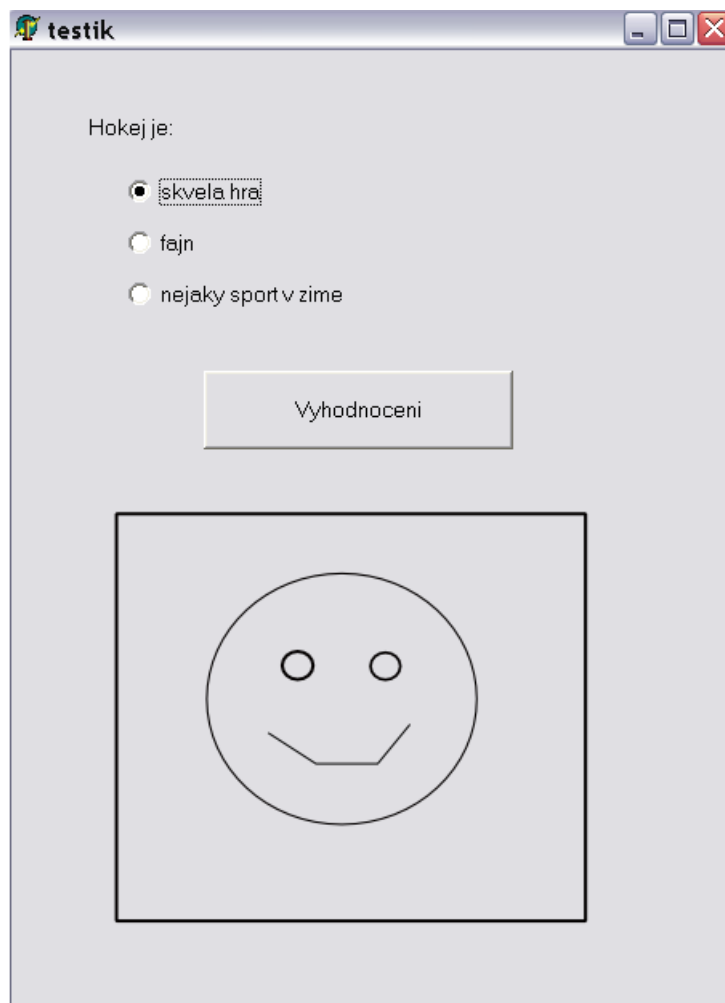


Obrázek 1: Program

3. pro zbylé dva buttony provedeme analogicky procedury `RadioButton2True`, `RadioButton3True`
4. vytvoříme `Image` na spodní část formuláře, do kterého budeme kreslit
5. vytvoříme button s popiskou `Vyhodnoceni` a propojíme ho s procedurou kresli na událost `OnClick`, pokud je zaškrtnutý `RadioButton1`, pak se provede vykreslení smajlíka, pro ostatní dle vlastní fantazie analogicky

```
if radioButton1.checked=true then
begin
  image1.canvas.Ellipse(50,50,240,240);
  image1.canvas.Ellipse(120,80,140,100);
  image1.canvas.Ellipse(170,80,190,100);

  image1.canvas.Polyline([Point(80, 130),
    Point(100,150), Point(140,170),
    Point(180,150),Point(200,130)]);
```



end;

5 Objekt Edit - editační okénko

Z editačních políček dostáváme typ string – řetězec, pro součet je tedy nutné provést konverzi na číslo pomocí funkce **VAL**. Výsledek pro zobrazení na obrazovce je nutné zase zpět převést na string pomocí funkce **STR**.

```
VAL(promenna string, promenna cislo,  
    chyba typu integer);  
STR(promenna cislo,string);
```

Příklad: Načteme z obrazovky dvě čísla z editačních boxů a do třetího vypíšeme po stisku tlačítka sečti jejich součet.

File → New Application

pro **Form1** nastavte

Properties

→ **Caption** → *Soucet cisel...* název formuláře

→ **Color** → *White* ... nastavení barvy pozadí

→ **Client Height, ClientWidth** ... šířka a délka formuláře

do daného formuláře umístíme tři editační políčka a jeden button

Standard

→ Edit - editační políčka, vložit třikrát

→ Button - tlačítko, vložit jednou

Nastavíme vlastnosti pro editační okna a button

Edit1

Properties

→ **Text** - prázdné

→ **Name** - *EditX*

Edit2

Properties

→ **Text** - prázdné

→ **Name** - *EditY*

Edit3

Properties

→ **Text** - prázdné

→ **Name** - *EditSoucet*

Button1

Properties

→ **Caption** - *SectiCisla*

→ **Name** - *SectiCisla*

Events

→ **OnClick** → napište zde *Soucet* a stiskněte Enter

- propojení tlačítka s procedurou Soucet

Dopíšeme zdrojový kód pro výpočet

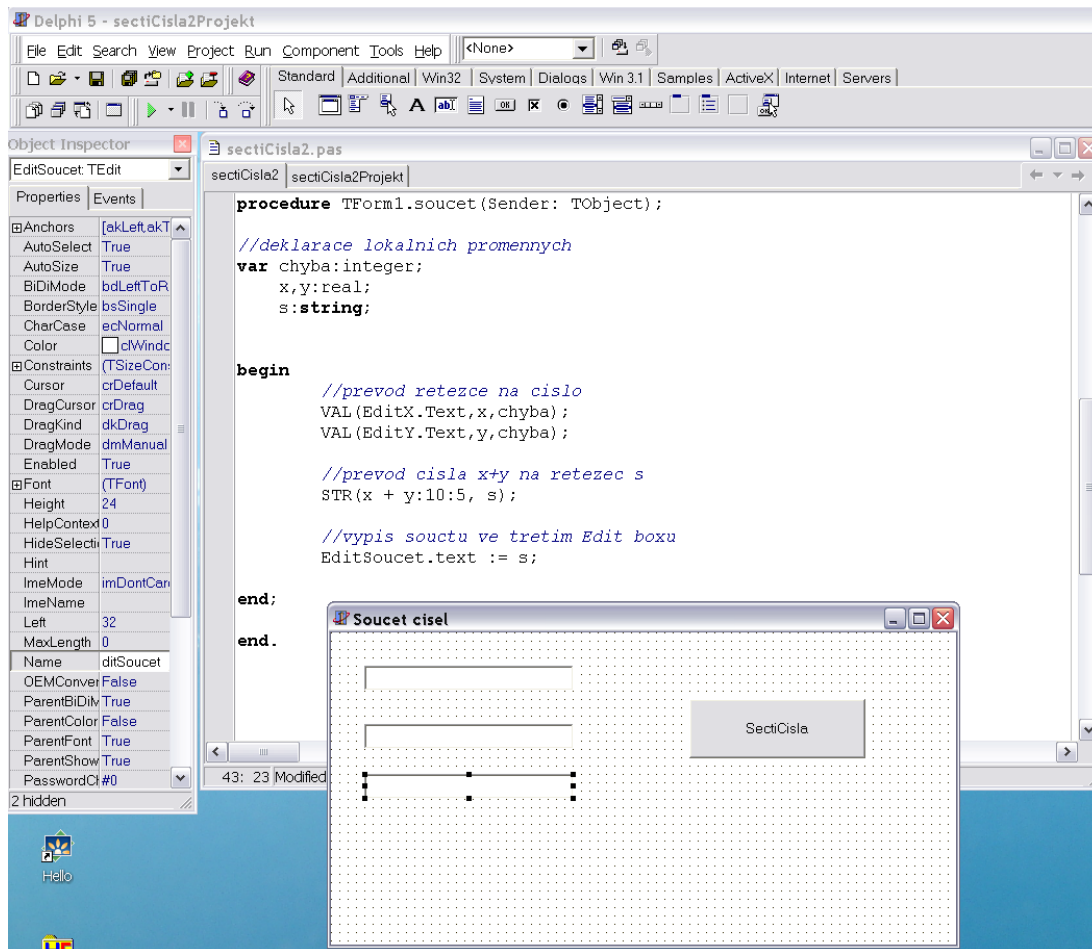
```
var  x,y: real;
      s:string;
      chyba:integer;

begin
  VAL(EditX.Text, x, chyba);
  VAL(EditY.Text, y, chyba);
```

```

STR(x+y:6:3,s);
EditSoucet.Text := s;
end;

```



Obrázek 2: Program

Příklad:

Zadají se dvě čísla do editačních okének a po stisku buttonu *kresli* se do image1 vykreslí obdélník o těchto stranách. Levý horní roh bude mít souřadnice (10,10).