

## Delphi – lekce 8

Vrátíme se k popisu často používaných komponent.

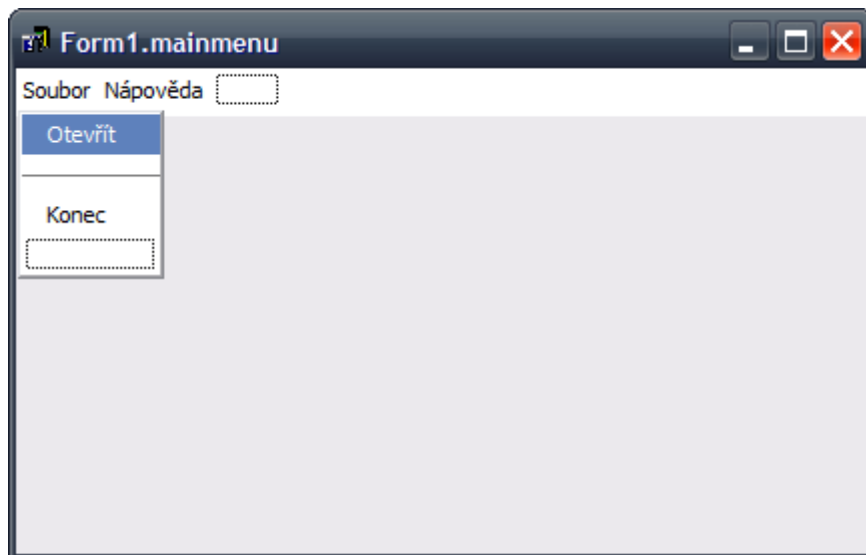
### TMainMenu

Komponenta symbolizuje hlavní menu aplikace se všemi podmenu. Po umístění komponenty na formulář se dá poklepáním na ikonu komponenty spustit návrhář. Obecně platí, že je výhodné kombinovat mainmenu s komponentou ActionList, která nám šetří práci, ale pro začátek se obejdeme bez její pomoci.

TMainMenu je vlastně takové zapouzdření pro komponenty **TMenuItem**, které symbolizují jednotlivé položky menu. Myslím, že už jsem psal, že je výhodné pro jednotlivé typy komponent používat různé prefixy (např. pro TLabel začínají komponenty na lbl, tj. lblName nebo lblNazev a k němu odpovídající TEditBox je edtName nebo edtNazev - podle mne zpřehledňuje kód, a jelikož používám rozšíření CnWizard, které mi generuje prefixy pro komponenty, tak mám ušetřenou práci).

Pokud nemáme podobné rozšíření, je dobré první vyplnit pro každou položku její název (**Name**) a pak až její **Caption**, jinak bude IDE podle zadané Caption generovat název. Tímto způsobem lze vygenerovat celé menu včetně podmenu.

Důležité: Pokud jako Caption zadáme – (tj. pomlčku), je položka menu změněna na oddělovač.



### Designer menu

Kliknutím na položku v designeru se vygeneruje patřičná obsluha OnClick, kde můžeme napsat svou obsluhu položky menu, např. pro uvedené Exit:

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

```
Application.Terminate;
end;
```

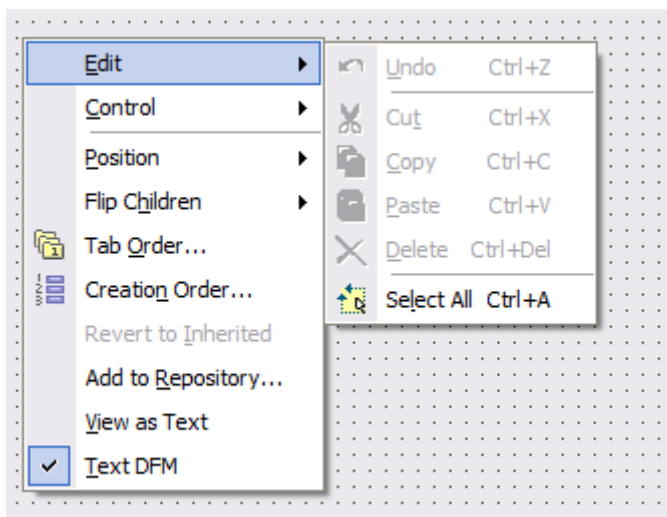
Pokud chceme u jednotlivých položek ikony menu, tak dáme na formulář nám již známou komponentu **TImageList** a přiřadíme ji do naší komponenty mainmenu. Pak jen u jednotlivých položek nastavíme ImageIndex (stejně jako u jiných komponent). Zajímavá je možnost nastavit klávesovou zkratku za pomoci vlastnosti **Shortcut** položky menu (a to ať v design módu nebo za běhu).

Již klasické property Enabled nebo Visible fungují jako u jiných komponent, navíc je zde property **Checked**, která u položky menu zobrazí zaškrtnutí.

Poslední věcí je možnost vlastního kreslení položek, což je trošku vyšší dívčí, ale umožňuje to velmi zajímavé věci. Možná někdy v budoucnu se k tomu vrátíme.

## TPopMenu

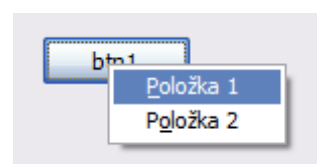
Tato komponenta je velmi podobná předešlé, jen s tím rozdílem, že může být na formuláři vícekrát a není ze začátku viditelná, jelikož symbolizuje popup neboli kontextové menu (tj. menu zobrazené kliknutím pravého tlačítka myši – viz. obrázek).



Položky se definují úplně stejně jako u TMainMenu. U komponenty na kterou chceme přidat naše popup menu jen nastavíme property PopupMenu. Program pak zaručí, že bude pro každou komponentu zobrazeno vybrané menu (v případě více rozdílných menu pro různé komponenty).

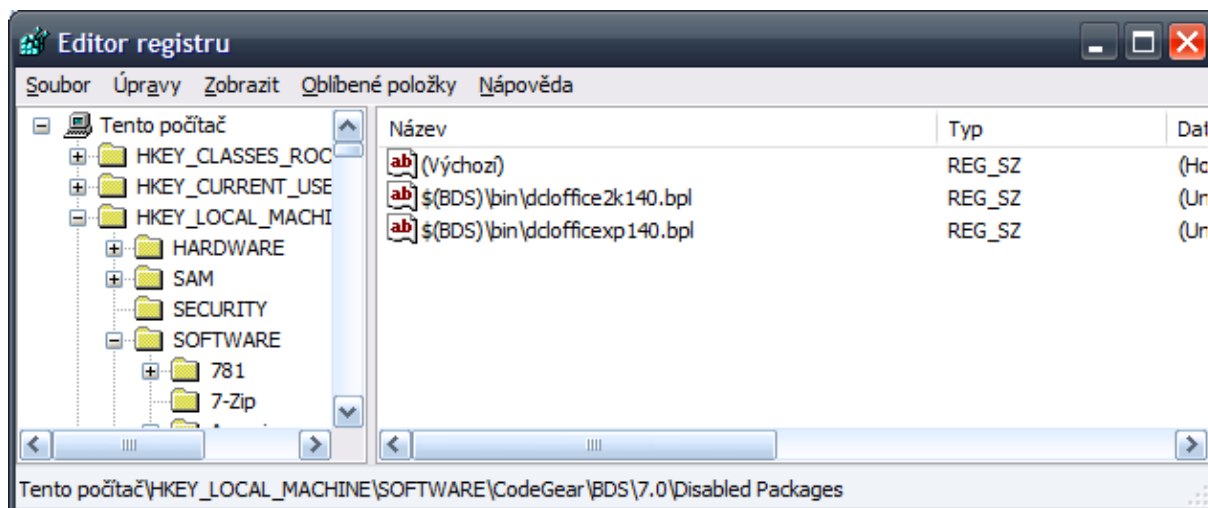
Lze také vyvolat zobrazení popmenu manuálně, např. po stisku tlačítka (TButton) myši. Předpokládejme, že máme na formuláři popmenu **pm1** a tlačítko. Pokud do obsluhy OnClick tlačítka napíšeme následující kód, bude na místě, kde je kurzor myši zobrazeno naše popmenu.

```
procedure TForm1.btn1Click(Sender: TObject);
begin
  pm1.Popup(Mouse.CursorPos.X, Mouse.CursorPos.Y);
end;
```

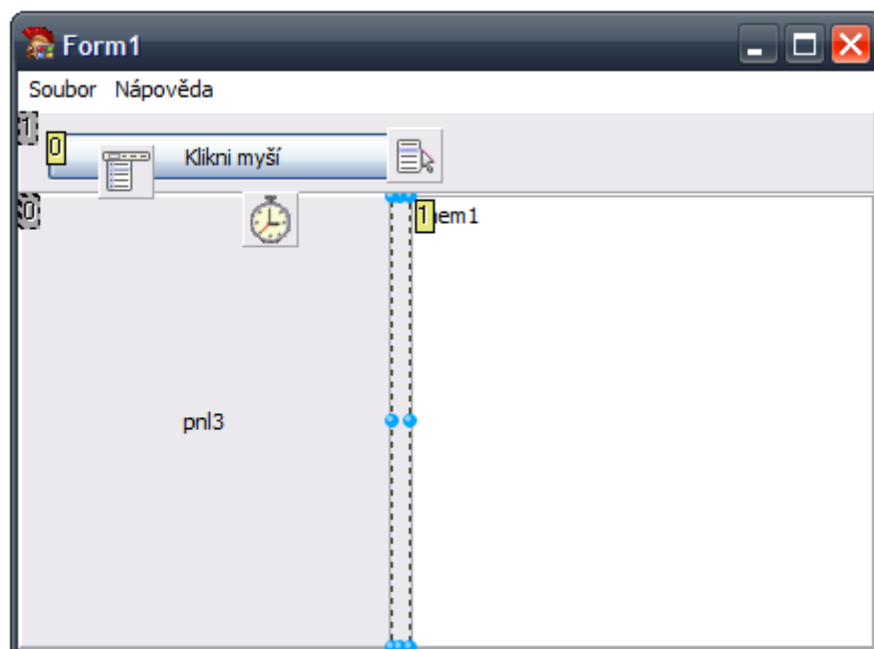


## TSplitter

Při popisu TPanel atd. jsem nějak pozapomněl na tuto důležitou komponentu. Slouží k rozdělení komponent právě typu TPanel (tj. nesou jiné komponenty), ale tak že uživatel může poměr rozdělení měnit přetažením myši. Špatně se to popisuje, přitom je to známá komponenta, např. v editoru registru je vpravo od kolmého scrollbaru a odděluje strom od položek a uživatel si poměr velikostí může určit.



V příkladu jsou následující komponenty:



- panel, který nejde vidět, ale jsou na něm panel **pnl3**, TMemo **mem1** a TSplitter mezi nimi
- první jsem vložil panel pnl3 a nastavil mu Align na alLeft

- následně vložil TSplitter, který má Align na alLeft automaticky, takže se přilepil zprava na panel
- nakonec jsem vložil TMemo (může být cokoliv) a nastavil align na alClient
- poslední věcí bylo, že jsem u splitteru nastavil šířku (Width) na 10, uživatelé se budou lépe myší strefovat

## TTimer

Poslední komponenta pro dnešek je na paletě v záložce Systém a jedná se o časovač.

Jedná se o velmi jednoduchou komponentu jen se dvěma property a jednou událostí, ale je velmi užitečná.

Property jsou **Interval** což je čas jak často se bude vyvolávat událost, **Enabled** určuje zda je časovač aktivní a událost **OnTimer** je vyvolávaná událost.

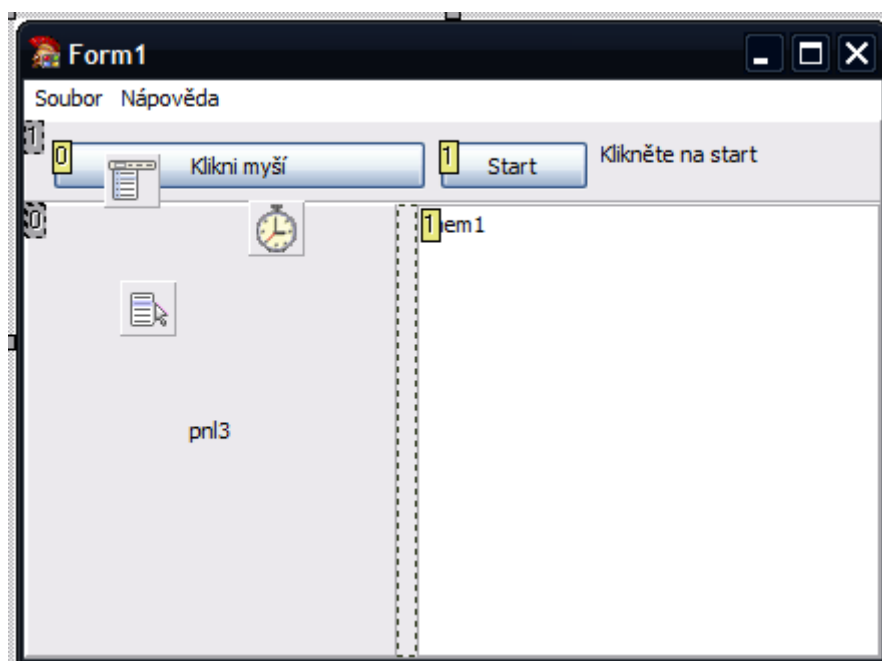
Do našeho předchozího příkladu jsem přidal časovač **tmr1**, Label **lblTimer** a tlačítko **btnStart**. U časovače jsem nastavil interval na 100 (ms) a Enabled na false (časovač není aktivní). Do obsluhy časovače jsem dal:

```
procedure TForm1.tmr1Timer(Sender: TObject);
begin
  lblTimer.Caption := DateTimeToStr(Now);
end;
```

tj. každých 100ms se do přidaného labelu aktualizuje aktuální čas. Poslední věc je ještě v obsluze tlačítka povolit časovač.

Kód je v příloze.

..



Domácí úkol:

- upravit poslední příklad i pro zastavování (s aktualizací tlačítka o text STOP)
- do hlavního menu přidat položku menu, která bude po kliknutí střídavě povolovat a zakazovat položku menu Exit (první klik zakáže, druhý povolí atd)
- přidat ještě jeden panel (viz. obr) pod pnl3 a mem1, který byl oddělen od obou jedním splitterem (původní jsou 1 a 2, nový panel je 3)

$$\begin{array}{r} 1 \mid 2 \\ \hline 3 \end{array}$$