

Číslo a název šablony	III/2 Inovace a zkvalitnění výuky prostřednictvím ICT
Číslo didaktického materiálu	EU-OPVK-VT-III/2-ŠR-314
Druh didaktického materiálu	DUM
Autor	RNDr. Václava Šrůtková
Jazyk	čeština
Téma sady didaktických materiálů	Programování v C# v příkladech III
Téma didaktického materiálu	Designer, soubory projektu
Vyučovací předmět	Seminář z informatiky
Cílová skupina (ročník)	Žáci ve věku 17–18 let
Úroveň žáků	Středně pokročilí
Časový rozsah	1–2 vyučovací hodiny
Klíčová slova	Designer, program, metoda Main, Controls
Anotace	Studenti se seznamují se strukturou projektových souborů, jejich metodami a vazbami. Získané poznatky aplikují při programování proměnlivého uživatelského rozhraní.
Použité zdroje	<p>ELLER, Frank. <i>C# - začínáme programovat: podrobný průvodce začínajícího uživatele</i>. 1. vyd. Praha: Grada, 2002, 240 s. ISBN 80-247-0324-6.</p> <p>OCHRANOVÁ, Renata a Michal KOZUBEK. <i>Objektově orientované programování v Turbo Pascalu</i>. 1. vyd. Brno: Masarykova univerzita, 1993, 117 s. ISBN 80-210-0659-5.</p> <p>VYSTAVĚL, Radek. <i>Moderní programování: sbírka úloh k učebnici pro středně pokročilé</i>. 1. vyd. Ondřejov: moderníProgramování, 2008-2009, 2 sv. ISBN 978-80-903951-3-8.</p> <p>VYSTAVĚL, Radek. <i>Moderní programování: učebnice pro pokročilé</i>. Ondřejov: moderníProgramování, 2011, 149 s. ISBN 978-80-903951-7-6.</p> <p>VYSTAVĚL, Radek. <i>Moderní programování: učebnice pro středně pokročilé</i>. Ondřejov: moderníProgramování s.r.o, 2008. ISBN 978-80-903951-2-1.</p>
Typy k metodickému postupu učitele, doporučené výukové metody, způsob hodnocení, typy k individualizované výuce apod.	<p>Text je možno využít ke společné práci, samostatné přípravě studentů, domácímu studiu apod.</p> <p>Při společné práci je vhodné během prezentace prohlížet kódy jednotlivých projektových souborů.</p> <p>V pracovním listu je zadání cvičení – většinou se</p>

	<p>jedná o úlohy, které by měli studenti naprogramovat samostatně. Není nutné, aby všichni zpracovali všechno, vhodné je diferencovat podle jejich zájmu a schopností. Součástí materiálu je zdrojový kód těchto příkladů.</p> <p>Návrh způsobu hodnocení: ohodnocení samostatné práce během hodiny např. podle volby a počtu úloh a elaborace řešení (efektivnost, komentáře...).</p>
--	--

Metodický list k didaktickému materiálu

Prohlášení autora

Tento materiál je originálním autorským dílem. K vytvoření tohoto didaktického materiálu nebyly použity žádné externí zdroje s výjimkou zdrojů citovaných v metodickém listu.

Obrázky (schémata a snímky obrazovek) pocházejí od autora.

314. Designer

Na jednoduchém příkladu si ukážeme, jak funguje Designer, jak vypadá kód, který generuje pro uživatelské rozhraní a jak můžeme vytvořit uživatelské rozhraní, které se bude měnit za běhu programu.

Připravíme si (další varianta) slide-show: při tiknutí časovače se v okně budou střídat obrázky umístěné v adresáři projektu bin/debug. Na formulář umístíme pouze tlačítko, které bude střídavě vypínat a zapínat časovač, interval časovače nastavíme na 1000 ms.

Obrázky pojmenujeme 1.BMP, 2.BMP..., aby se mohly střídat automaticky.

Výpis Form1.cs

```
namespace _47_Designer
{
    public partial class Form1 : Form
    {
        int i=1; //číslo obrázku

        public Form1()
        {
            InitializeComponent();
        }
    }
}
```

```

private void timer1_Tick(object sender, EventArgs e)
{
    i++;
    Refresh();
    if (i ==4) i = 0;
}

private void buttonSS_Click(object sender, EventArgs e)
{
    timer1.Enabled = !(timer1.Enabled);
}

private void Form1_Paint(object sender, PaintEventArgs e)
{
    Graphics kp = e.Graphics;
    Image obr=Image.FromFile(i.ToString()+".BMP");
    kp.DrawImage(obr, 20, 20);
}
}
}

```

Jak vypadá kód Designeru – tedy

Form1.Designer.cs

```

namespace _47_Designer
{
    partial class Form1
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

```

```
/// <summary>
/// Clean up any resources being used.
/// </summary>
/// <param name="disposing">true if managed resources should be disposed; otherwise,
false.</param>
protected override void Dispose(bool disposing)
{
    if (disposing && (components != null))
    {
        components.Dispose();
    }
    base.Dispose(disposing);
}

#region Windows Form Designer generated code

/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
private void InitializeComponent()
{
    this.components = new System.ComponentModel.Container();
    this.timer1 = new System.Windows.Forms.Timer(this.components);
    this.buttonSS = new System.Windows.Forms.Button();
    this.SuspendLayout();

    //
    // timer1

```

```
//  
this.timer1.Interval = 1000;  
this.timer1.Tick += new System.EventHandler(this.timer1_Tick);  
//  
// buttonSS  
//  
this.buttonSS.Location = new System.Drawing.Point(86, 235);  
this.buttonSS.Name = "buttonSS";  
this.buttonSS.Size = new System.Drawing.Size(75, 23);  
this.buttonSS.TabIndex = 0;  
this.buttonSS.Text = "Strat/Stop";  
this.buttonSS.UseVisualStyleBackColor = true;  
this.buttonSS.Click += new System.EventHandler(this.buttonSS_Click);  
//  
// Form1  
//  
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);  
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;  
this.ClientSize = new System.Drawing.Size(292, 270);  
this.Controls.Add(this.buttonSS);  
this.Name = "Form1";  
this.Text = "Slide-Show";  
this.Load += new System.EventHandler(this.Form1_Load);  
this.Paint += new System.Windows.Forms.PaintEventHandler(this.Form1_Paint);  
this.ResumeLayout(false);  
}  
#endregion
```

```
private System.Windows.Forms.Timer timer1;

private System.Windows.Forms.Button buttonSS;

}

}
```

Všimněte si, především, že oba výpisy (Jak Form1.cs tak Designer.cs) začínají definicí třídy: **partial class Form1**. Slovo partial znamená, že je tato definice rozdělena do více textů. – zde **Form1.cs** a **Form1.Designer.cs**

Popíšeme si některé části kódu.

```
private void InitializeComponent()
```

```
{...}
```

Metoda, ve které jsou definovány členské proměnné pro odkazy na formulář, tlačítko a časovač. Vytvoří se voláním příslušných konstruktorů a nastaví se jejich vlastnosti.

Pro tlačítko:

```
this.buttonSS = new System.Windows.Forms.Button(); – konstruktor
```

```
this.buttonSS.Location = new System.Drawing.Point(86, 235); – umístění
```

(Location je instance třídy Point a udává levý horní roh objektu)

```
this.buttonSS.Name = "buttonSS";
```

```
this.buttonSS.Size = new System.Drawing.Size(75, 23);
```

```
this.buttonSS.TabIndex = 0;
```

```
this.buttonSS.Text = "Strat/Stop";
```

```
this.buttonSS.UseVisualStyleBackColor = true;
```

Připojení obslužné metody události Click:

```
this.buttonSS.Click += new System.EventHandler(this.buttonSS_Click);
```

Vložení ovládacích prvků do okna:

(nachází se v sekci okna (formuláře)

```
this.Controls.Add(this.buttonSS);
```

Controls je seznam všech ovládacích prvků okna.

Vraťme se na začátek výpisu **Form1.cs**

```
public Form1()
{
    InitializeComponent();
}
```

Metoda se jmenuje stejně jako třída – je to tedy konstruktor, který volá metodu `InitializeComponent()`;

Tyto dva zdrojové texty tedy definují třídu `Form1` – okno programu. Její instance se vytváří v dalším zdrojovém kódu – **Program.cs**.

```
namespace _47_Designer
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles(); // vzhled viditelných prvků
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1()); //vytvoření okna v metodě Application.Run
        }
    }
}
```

V tomto kódu se definuje metoda **Main** – vstupní bod každého programu v jazyce C#. **Její první příkaz je ten, který program vykoná po spuštění.**

Metoda `Application.Run`

- Zobrazí okno programu

- Čeká v tzv. **smyčce událostí**, ta přebírá od operačního systému události jako stisk klávesy, pohyb myši, tik časovače apod., dokud touto událostí není zavření okna. Tato smyčka řídí běh událostí v programech s grafickým interfacem.

Měnící se uživatelské rozhraní – vložení komponenty za běhu programu

Využijeme nové znalosti k vložení tlačítka do místa, kde uživatel klepne myší. Každé tlačítko bude mít své číslo a při kliknutí na tlačítko se zobrazí zpráva AHOJ.

```
int cislo = 1;

public Form1()

private void tlacitko_Click(object sender, EventArgs e)

{

    MessageBox.Show("Ahoj");

}

private void Form1_MouseDown(object sender, MouseEventArgs e)

{

    int x = e.X;

    int y = e.Y;

    Button tlacitko = new Button();

    tlacitko.Location = new Point(x, y);

    tlacitko.Size = new Size(40, 20);

    tlacitko.Text = cislo.ToString();

    tlacitko.Click+=new EventHandler(tlacitko_Click);

    Controls.Add(tlacitko);

    cislo++;

}
```

Důležité

Definice **partial class Form1**. Slovo partial znamená, že je tato definice rozdělena do více textů. – **Form1.cs** a **Form1.Designer.cs**

private void InitializeComponent()

```
{...}
```

Metoda, ve které jsou definovány členské proměnné pro komponenty

Vložení ovládacích prvků do okna:


```
Controls.Add(prvek);
```

Controls je seznam všech ovládacích prvků okna

Tyto dva zdrojové texty tedy definují třídu Form1 – okno programu. Její instance se vytváří v dalším zdrojovém kódu – **Program.cs**.

```
static void Main()  
{
```

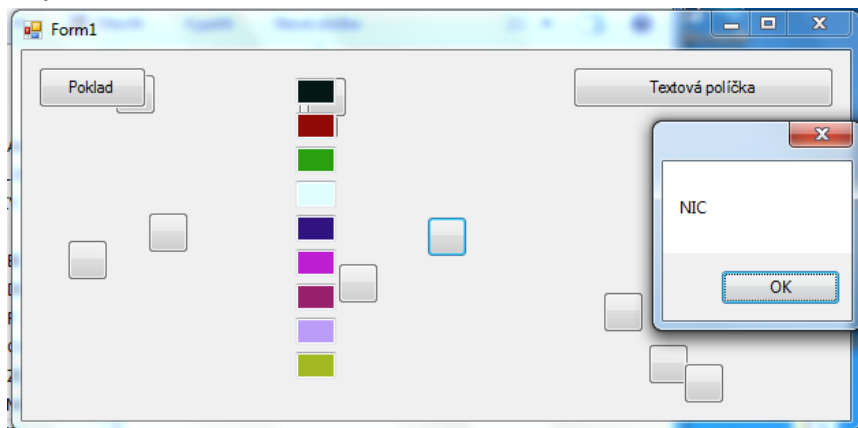
Metoda **Main** – vstupní bod každého programu v jazyce C#.

Pracovní list

Cvičení

Odpovězte na otázky:

1. Proč definice třídy okna obsahuje slovo partial?
2. Kde jsou definovány členské proměnné pro komponenty?
3. Jak se nazývá seznam všech ovládacích prvků okna?
4. Jakou metodou se do něj přidává?
5. Co je to metoda Main?



Příklad 1.

Připravte si formulář, na kterém budou dvě tlačítka – Poklad a Textová políčka.

Při stisknutí tlačítka Textová políčka se objeví pod sebou zarovnaná řada textových políček s náhodnými barvami pozadí.

Při stisknutí druhého tlačítka se zobrazí náhodně rozhozená tlačítka bez textu. Pod jedním z nich je schován poklad – Zpráva **poklad**, jinak zpráva **nic**.

Řešení

Odpovědi:

1. Proč definice třídy okna obsahuje slovo partial?
Definice je rozdělena do textů **Form1.cs** a **Form1.Designer.cs**

2. Kde jsou definovány členské proměnné pro komponenty?
Metoda **InitializeComponent()**
3. Jak se nazývá seznam všech ovládacích prvků okna?
Controls
4. Jakou metodou se do něj přidává?
Controls.Add
5. Co je to metoda Main?
Vstupní bod každého programu v jazyce C#.

```
Random nahoda = new Random();

    int s, v;

...

private void tlac_Click(object sender, EventArgs e)
{
    MessageBox.Show("NIC");
}

private void tlacP_Click(object sender, EventArgs e)
{
    MessageBox.Show("POKLAD");
}

private void Form1_Load(object sender, EventArgs e)
{
    s = ClientSize.Width;
    v = ClientSize.Height;
}

private void buttonPoklad_Click(object sender, EventArgs e)
{
    //hra poklad

    int x = nahoda.Next(10, s-40);

    int y = nahoda.Next(10, v - 40);

    Button tlacitko = new Button();

    tlacitko.Location = new Point(x, y);

    tlacitko.Size = new Size(30, 30);
}
```

```

tlacitko.Click += new EventHandler(tlacP_Click);

Controls.Add(tlacitko);

for (int i = 0; i < 9; i++)
{
    x = nahoda.Next(10, s - 40);
    y = nahoda.Next(10, v - 40);

    Button tlacit = new Button();

    tlacit.Location = new Point(x, y);

    tlacit.Size = new Size(30, 30);

    tlacit.Click += new EventHandler(tlac_Click);

    Controls.Add(tlacit);
}
}

private void buttonText_Click(object sender, EventArgs e)
{
    //řada barevných textových políček

    int r, g, b;
    int x = 200;
    int y = 20;
    for (int i = 0; i < 9; i++)
    {
        r = nahoda.Next(256);
        g = nahoda.Next(256);
        b = nahoda.Next(256);
        TextBox Pole = new TextBox();
        Pole.Location = new Point(x, y);
        Pole.Size = new Size(30, 200);
        Pole.BackColor = Color.FromArgb(r, g, b);
        Controls.Add(Pole);
        y += 25;
    }
}

```