Jazyk C# (seminář 7)

Pavel Procházka

KMI

5. listopadu 2014

WindowsForms basics

- Windows forms je abstrakce nad WinApi napsaném v C
- Hello world aplikace ve WinAPI má 114! řádků podle `http://msdn.microsoft.com/cs-cz/library/bb384843.aspx`
- Dobrá zpráva je, že WindowsForms má daleko méně řádků a hlavně méně složitých
- Navíc VisualStudio obsahuje UI Designer, který umožňuje celé GUI navrhnout graficky, tudíž není třeba ho otrocky kódovat
- VisualStudio ve spojení s UI designérem generuje automatizovaně kód, který v jistém smyslu usnadňuje psaní kódu, hlavně zlepšuje křivku učení
- VisualStudio za vás doplní eventhandlery, takže se psaní jednodušších programů, které nepotřebují budovat GUI za běhu redukuje na vyplnění kódu událostí
- Na druhou stranu pokud budete potřebovat budovat GUI za běhu, tak už Designer ztrácí na významu
- Není to příliš multiplatformní kvůli vazbám na WinAPI, i když mono to umí
- Monodevelop nemá podporu pro Windows Forms (pouze experimentální – `http://www.mono-project.com/archived/winforms_designer/`)

GUI aplikace v C#

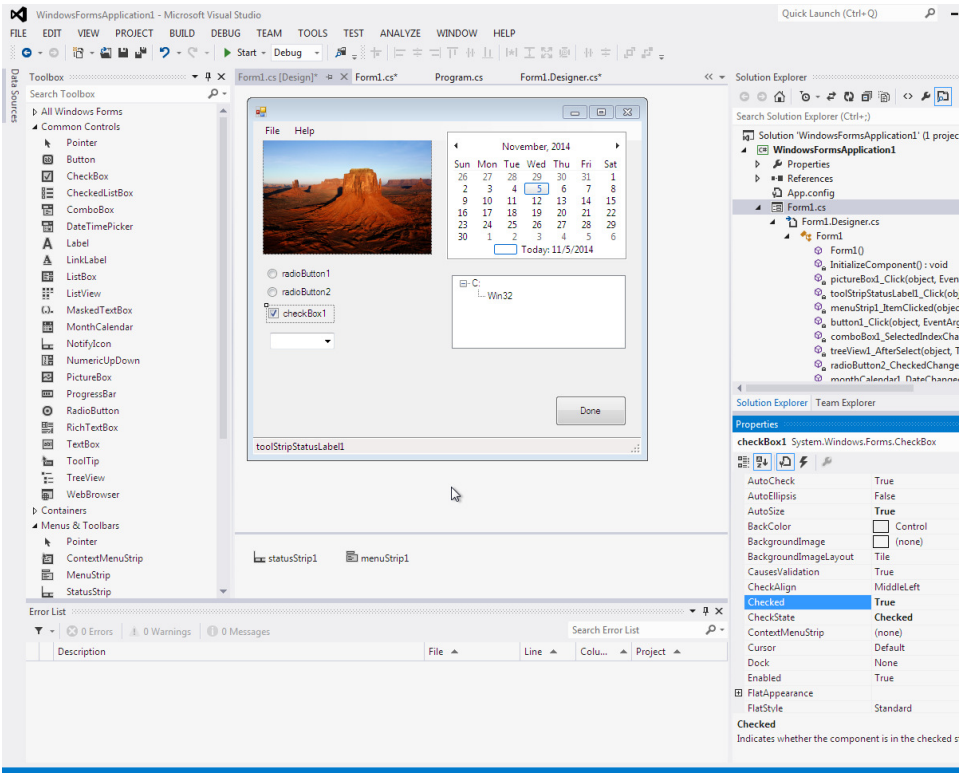
- Ve Visual Studiu nyní volíme Windows Forms Application – ne Console Application
- Program má stejnou strukturu jako Console Application

```
using System;
using System.Windows.Forms;

/*
 Form1 je trida, kterou jsme vytvorili
 pomoci designeru
 */

namespace WindowsFormsApplication1
{
    static class Program
    {
        static void Main()
        {
/* staticka trida System.Windows.Forms.Application */
            Application.Run(new Form1());
        }
    }
}
```

Seznámení s Designerem VS 2012



Důležité třídy a metody Windows Forms

- Form – třída reprezentující okno, od které musí každé okno dědit
- Metoda Form.InitializeComponent() – zařídí, že se okno začne kreslit
- Metoda u GUI prvků (třídy, co dědí od Windows.Forms.Component) Dispose – vymaže grafický prvek
- Každý GUI element je nějaká třída, takže máme Button, TreeNode, CheckBox, ComboBox, ProgressBar, Label, atd. (viz. MSDN)
- Lze vytvářet i vlastní GUI prvky děděním od třídy `Component` (se kterými pak ale nepůjde pracovat pomocí Designeru)
- Do některých grafických prvků (ty, co dědí `Control`) lze kreslit pomocí třídy `Graphics`, lze tedy docílit efektu jakéhosti plátna
- Samostatnou kapitolou jsou pak obrázky – `Bitmap`, `Image`

Co Designer produkuje

- Designer generuje kód na základě toho, co si „naklikáte“
- Výstupem je potom nějaký form (což je třída), která obsahuje další prvky, které jsou na něj navázány ve formě členských proměnných a jsou inicializovány, podle toho, co vyplníte v designeru
- Objekty tedy nejsou v žádé kolekci a nelze je inteligentne procházet
- Designer obsahuje všechny možné nastavitelné vlastnosti pro daný objekt a nabízí všechny eventy, které umožňuje jednoduše vyplnit – přenese uživatele na místo, kde může implementovat událost
- Časem ten kód ale začne být dosti nepřehledný

Designer – vygenerovaný kód

- Designer rozděluje implementaci třídy `Form` na několik souborů – to lze pomocí klíčového slova `partial` – deklaraci třídy tedy můžeme mít ve více souborech pomocí partial

```
/* vytvoreni a inicializace labelu ve tride Form1
this.label1 = new System.Windows.Forms.Label();
this.SuspendLayout();
this.label1.AutoSize = true;
this.label1.Location = new System.Drawing.Point(25,
this.label1.Name = "label1";
this.label1.Size = new System.Drawing.Size(31, 13),
this.label1.TabIndex = 0;
this.label1.Text = "Hello";
this.label1.Click += new System.EventHandler(this.
...
private System.Windows.Forms.Label label1;
```

Bitmapy a obrázky

- Součást `System.Drawing`
- `http://msdn.microsoft.com/cs-cz/library/System.Drawing(v=vs.110).aspx`
- Důležitá třída `Color`, inicializuje se pomocí `ARGB` bajtových hodnot
- Abstraktní třída `Image` od ní dědí/implementuje třída `Bitmap`
- Do bitmapy lze přímo kreslit a číst pixely pomocí `SetPixel()`, `GetPixel()`, což je docela asi nejrychlejší softwarová metoda v C#, pro manipulaci s pixely
- Třída `Graphics` zajišťuje kreslení do elementu, který dědí `Control`, objekt graphics získáme metodou `CreateGraphics();`
- Pomocí metod třídy `Graphics` pak můžeme kreslit mnoho různých tvarů (GDI+)

Příklad kreslení do bitmapy per pixel

```
Bitmap b = new Bitmap( iw , ih);
/* pictureBox1 je vytvoren Designerem */
pictureBox1.DrawToBitmap(b,
    new Rectangle(0,0,iw, ih) );

for (int y = 0; y < ih; y++)
{
    for (int x = 0; x < iw; x++)
    {
        Color lc = b.GetPixel(x, y);
        b.SetPixel(x, y, Color.FromArgb(lc.R, 0, 0));
    }
}

pictureBox1.Image = b;
```

Výsledek kreslení

Kreslení grafických primitiv

- Třída `Graphics` obsahuje optimalizované kreslení primitiv, navíc kreslí automaticky, jak je to možné
- `DrawLine` – kreslí úsečku
- `DrawArc` – kreslí elipsu
- `DrawRectangle` – kreslí obdélník
- `DrawPolygon` – polygon
- `DrawImage` – obrázek
- `Fill*` – vyplňuje daná primitiva
- `Pen` – kreslí čáru různým stylem (tečkovaně, šrafovaně, ...), různou barvou a různou tloušťkou
- `Brush` – abstraktní třída reprezentující vyplňovací „štětec“
- `SolidBrush` – jednobarevný štětec, `TextureBrush` – vyplňuje bitmapou, `GradientBrush` – vykresluje gradientem

Kreslení grafických primitiv

```
//this.pictureBox1.Dispose();
int iw = pictureBox1.Width;
int ih = pictureBox1.Height;
Color c = Color.FromArgb(255, 0, 255);
Pen p = new Pen(c, 8f);
Graphics g = this.pictureBox1.CreateGraphics();
g.DrawRectangle(p, new Rectangle(0, 14, iw, ih-16));
```

Jednoduché textové dialogy

- Často chceme informovat uživatele – k tomu slouží jednoduché textové dialogy
- Ty umí vyvolat statická třída `MessageBox`
- Metoda `Show` vytiskne text
- MessageBoxy jsou modální – uživatel je musí odklepnout, před další prací
- Analogie ke `Console.WriteLine()` – tu muzeme pouzivat i tak, ale bude se tisknout do konzole!

```
MessageBox.Show("Hello world");
```

Odkazy

- Už umíte prakticky celou syntaxi C#, tudíž by neměl být problém rozumět dokumentaci
- Pěkný tutoriál najdete kupodivu na stránkách mono `http://zetcode.com/gui/csharpwinforms/`
- Tutoriál je i na MSDN `http://msdn.microsoft.com/en-us/library/zftbwa2b(v=vs.110).aspx`
- Samozřejmostí je referenční dokumentace `http://msdn.microsoft.com/en-us/library/system.windows.forms(v=vs.110).aspx`
- `http://msdn.microsoft.com/en-us/library/system.drawing(v=vs.110).aspx`

A teď vy

- Vykreslete vlajku EU a po kliknutí na nějakou hvězdičku se objeví nějaký text formou MessageBoxu