

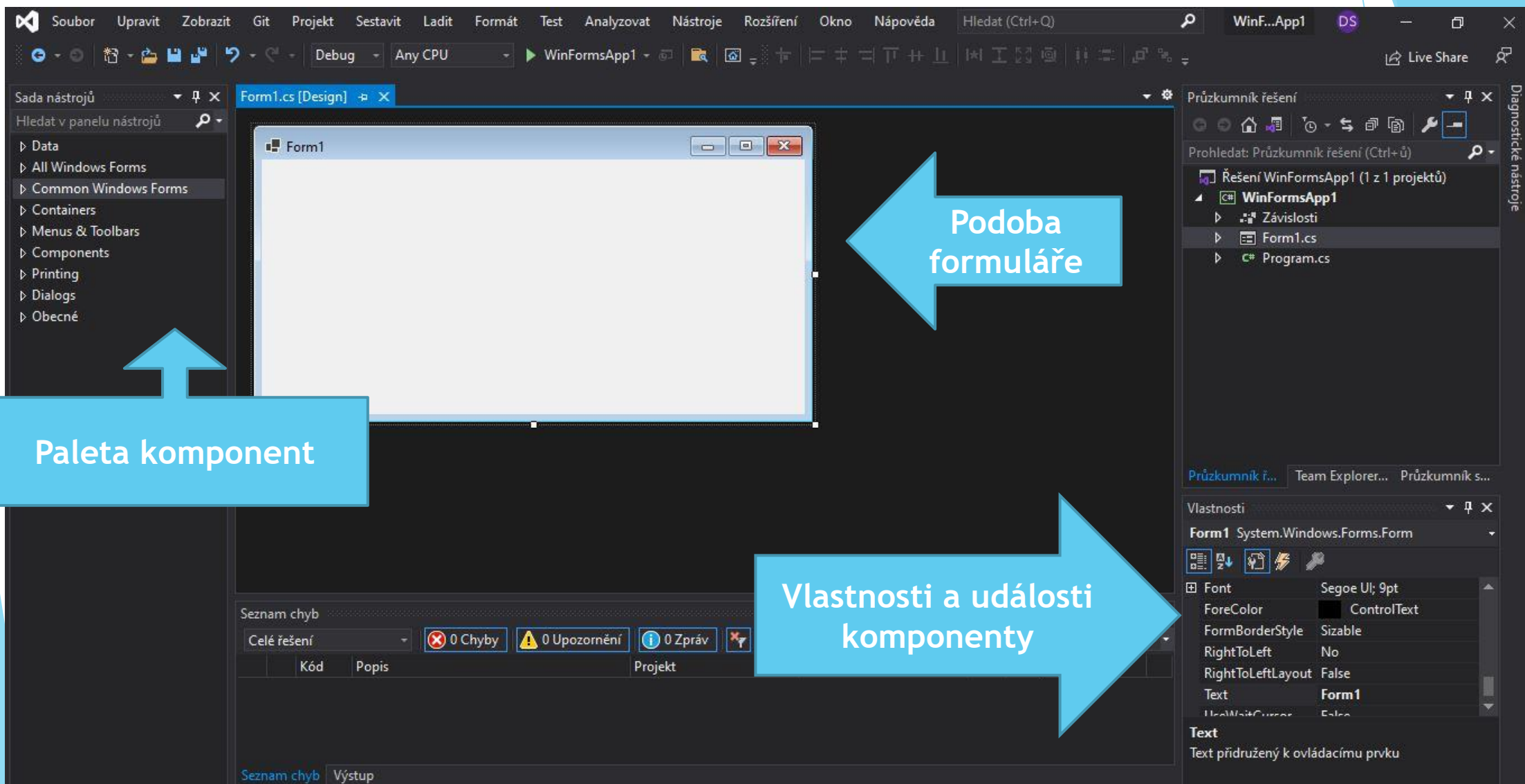
Programování

Formulářové aplikace

Windows Form Application (WFA)

- ▶ Framework umožňující vytvářet jednoduché desktopové aplikace
- ▶ Návrh formulářové aplikace nám umožňuje tzv. DESIGNER
- ▶ Široká paleta formulářových prvků (komponent):
 - ▶ Vstupní výstupní prvky
 - ▶ Ovládací prvky - tlačítka, checkboxy, ...
- ▶ Technologie vyvíjena posledních 20 let
 - ▶ Široká podpora, jednodušší na vytváření aplikací, rastrová grafika
- ▶ Postupně lehce ustupuje modernějšímu WPF (Windows Presentation Form)
- ▶ WPF pro začátečníka náročnější - vyžaduje znalost XAML
- ▶ WPF využívá téměř všechny koncepty WFA, které pak při přechodu lze dobře uplatnit

První pohled do nového WFA projektu



Grafický návrhář WFA

- ▶ Návrhář slouží k vizualizaci výsledné aplikace a rozmístění jednotlivých prvků
- ▶ Součástí návrháře je paleta komponent a zdrojový kód, generující výslednou podobu formuláře (NazevProjektu.Designer.cs)
- ▶ Komponenty na formuláře můžeme vkládat pomocí drag and drop přímo z palety nástrojů
- ▶ Obsahuje inicializaci jednotlivých komponent a operace nad komponentami
- ▶ Visual studio si tyto inicializace generuje samo, vývojář má o to méně práce
 - ▶ Všechny vlastnosti, které komponentám nastavujeme se zde generují jako hodnoty atributů
- ▶ Tento soubor se nedoporučuje manuálně upravovat

Grafický návrhář WFA

```
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;
    Počet odkazů: 0
    protected override void Dispose(bool disposing)
    {
        if (disposing && (components != null))
        {
            components.Dispose();
        }
        base.Dispose(disposing);
    }
}

Windows Form Designer generated code
```

Partial označuje rozdělení do více souborů

Obslužná metoda pro úklid

Kód generovaný designerm

Grafický návrhář implementace

- ▶ Do implementace kódu formuláře se můžeme dostat skrze kontextové menu nebo pomocí klávesy F7

```
using System.Windows.Forms;

namespace WinFormsApp1
{
    Počet odkazů: 3
    public partial class Form1 : Form
    {
        Počet odkazů: 1
        public Form1()
        {
            InitializeComponent();
        }

        Počet odkazů: 1
        private void Form1_Load(object sender, EventArgs e)
        {
        }
    }
}
```

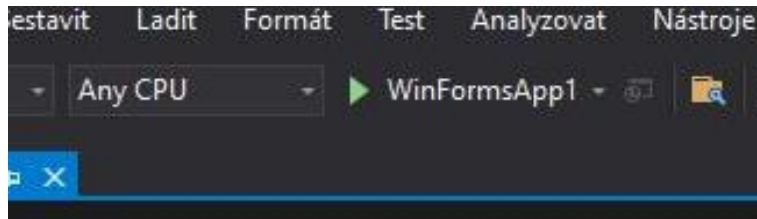
Rozšíření původní třídy Form

Konstruktor našeho formuláře

Funkce události načtení formuláře

Spuštění WFA projektu

- ▶ Pro spuštění můžeme využít klávesové zkratky CTRL + F5 (ladění)
- ▶ Spuštění pomocí zeleného PLAY tlačítka v menu pro ostrou verzi



- ▶ Stejně jako konzolová aplikace i zde se nachází hlavní funkce v souboru Program.cs
- ▶ U nového projektu řeší pouze inicializaci hlavního okna a jeho spuštění

```
6
7 namespace WinFormsApp1
8 {
9     Počet odkazů: 0
10    static class Program
11    {
12        /// <summary>
13        /// The main entry point for the application.
14        /// </summary>
15        [STAThread]
16        Počet odkazů: 0
17        static void Main()
18        {
19            Application.SetHighDpiMode(HighDpiMode.SystemAware);
20            Application.EnableVisualStyles();
21            Application.SetCompatibleTextRenderingDefault(false);
22            Application.Run(new Form1());
23        }
24    }
25 }
```

Komponenty WFA

- ▶ Formulářové aplikace využívají koncepce OOP
- ▶ Každá komponenta (objekt) je definována:
 - ▶ Vlastnostmi (atributy)
 - ▶ Metodami
 - ▶ Událostmi
- ▶ Události jsou vyvolané uživatelem - např. kliknutí na tlačítko
- ▶ Událost je reprezentována jako obslužná metoda přijímající dva parametry
 - ▶ **Object sender** - poskytuje odkaz na komponentu, se kterou pracujeme
 - ▶ **EventArgs e** - předává specifické vlastnosti pro obslužní metodu
- ▶ Tyto události můžeme vygenerovat pomocí učinění této akce nebo z panelu vlastností a událostí

