

COM – Práce s objektovým modelem jiné aplikace

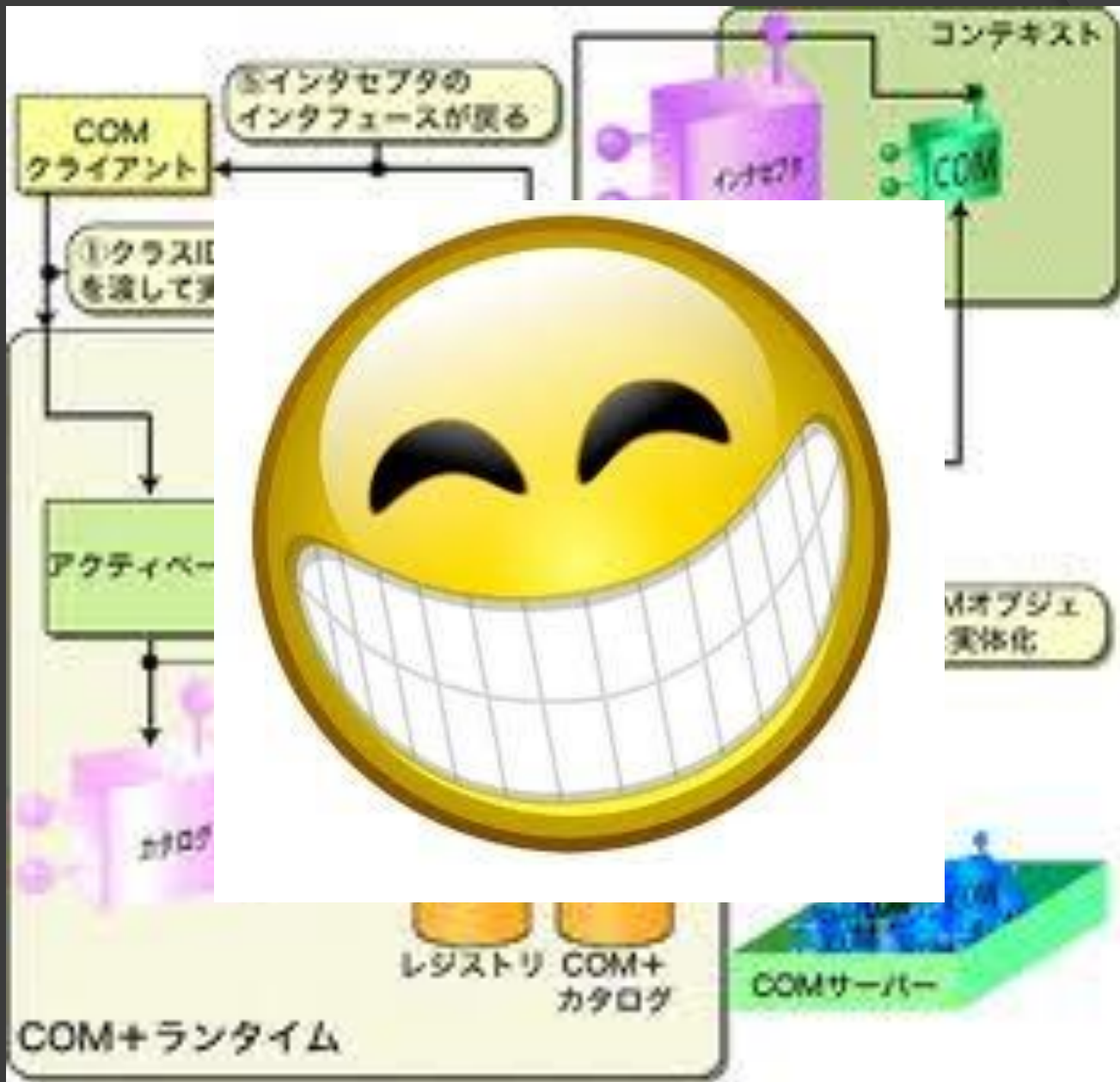
**Centrum pro virtuální a moderní metody a formy vzdělávání na
Obchodní akademii T.G. Masaryka, Kostelec nad Orlicí**



Co je to COM ?



- ⦿ ***Component Object Model***
- ⦿ Technologie umožňující komunikaci SW komponent na OS Windows
- ⦿ Znovupoužitelné části kódu, které může využívat více programů
- ⦿ COM objekty mohou být napsány v různých jazycích
- ⦿ COM technologie zahrnují COM+, Distributed COM (DCOM) and ActiveX® Controls.
- ⦿ Microsoft poskytuje COM interface pro mnoho Windows aplikací
- ⦿ COM je použit i u Microsoft Office. Např. COM OLE technologie umožňuje Word dokumentům dynamicky linkovat data z Excelu atd.



- Pohádka o třech bratřech (sousedech)
- Co kdyby měli jeden traktor dohromady ?
- Jaké by to mělo výhody a nevýhody?





Proč interface ?

Interface
Metoda VypoctiObsah
(tady není žádná
definice)

Class ctverec

Metoda VypoctiObsah
(Konkrétní výpočet)

Class kruh

Metoda VypoctiObsah
(Konkrétní výpočet)

Class trojuhelnik

Metoda VypoctiObsah
(Konkrétní výpočet)

● **Main hloupě:**

```
ctverec c = new ctverec;  
kruh k = new kruh;  
trojuhelnik t = new trojuhelnik;  
Celkem = k.VypoctiObsah() +  
t.VypoctiObsah() + c.VypoctiObsah();
```

A jak by to bylo třeba v kolekci ? a cyklu for each

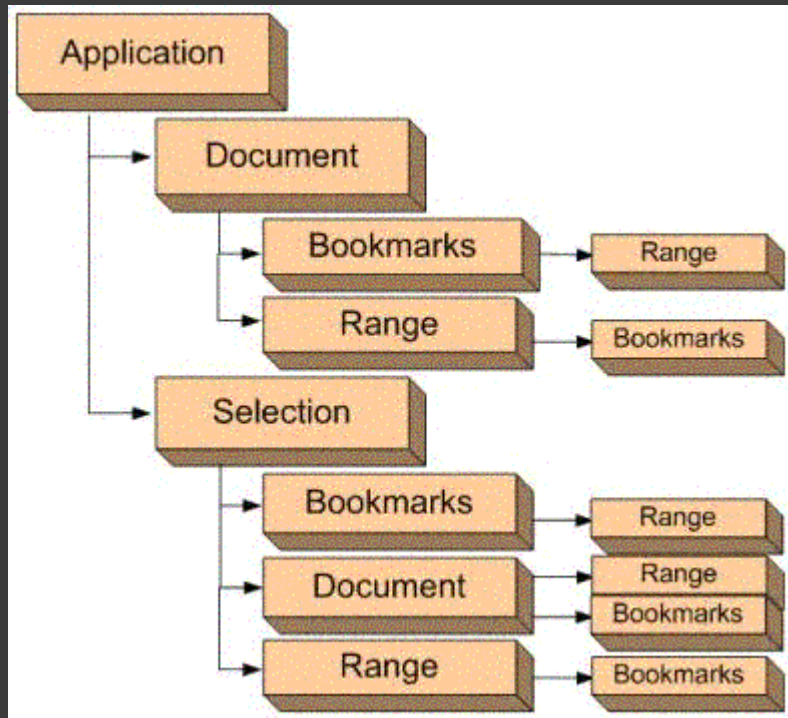
tvar[i].VypoctiObsah() ...kde tvar[i] - tvar je instancí našeho rozhraní, na indexu i v poli tvar je uložena nová instance libovolného objektu (např new Kruh()).

Ukázkový příklad: COM_interface_Demo_02



Aplikace poskytují COM objektový model vývojářům, tak aby ho mohli využít ve svých aplikacích

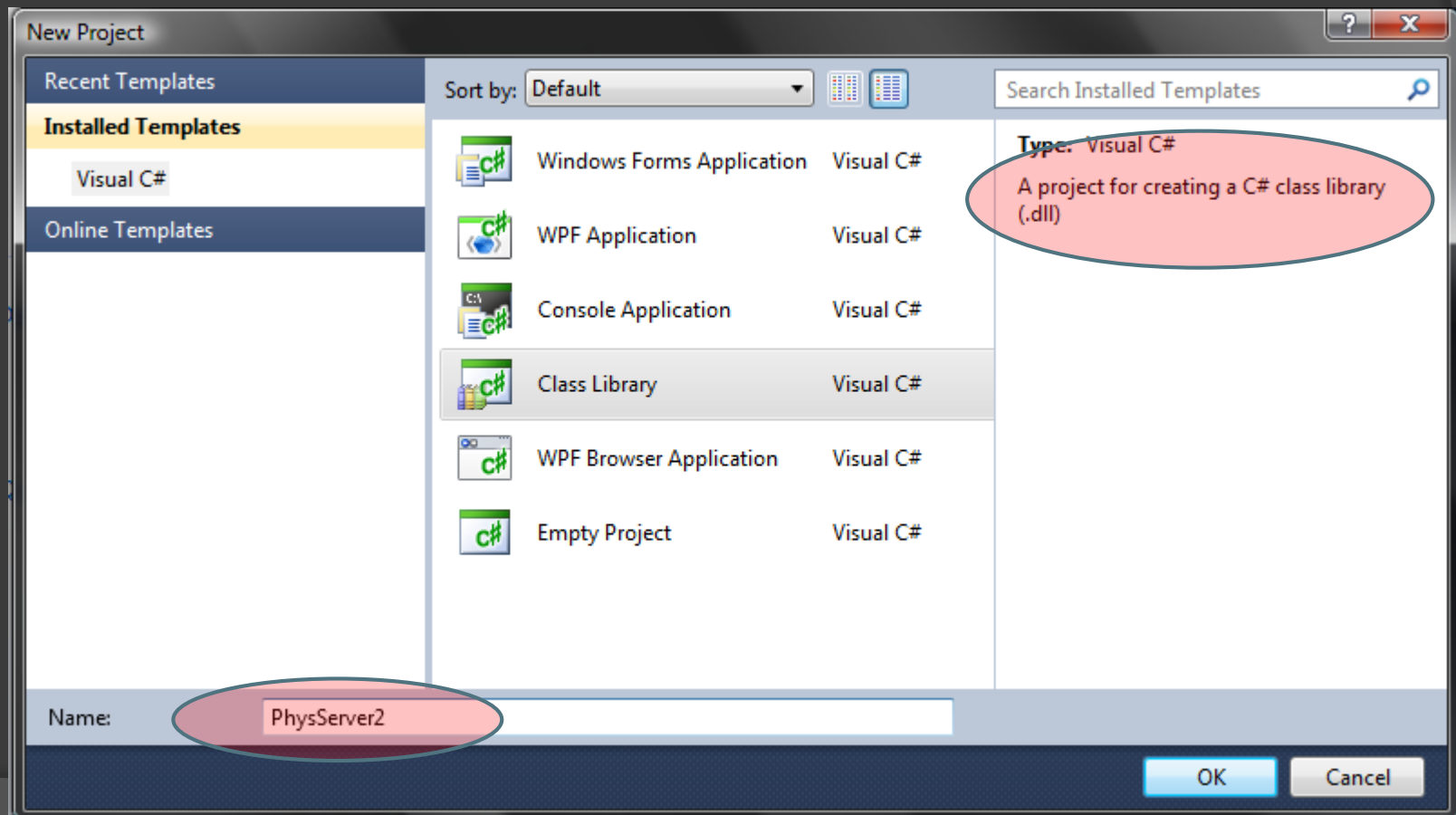
● Příklad COM modelu MS Word:



Step by Step – vytvoření a použití vlastní .dll knihovny

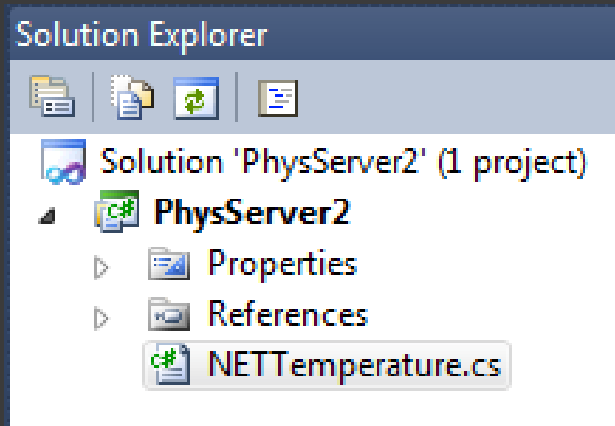


- Vytvořte nový projekt typu ClassLibrary a nazvěte ho **PhysServer2**





- *Class1.cls* přejmenujte na ***NETTemperature.cls***



- Otevřete textový soubor ***CODE_OLE_NETTemperature_Demo.txt*** a zkopírujte obsah souboru namísto kódu třídy *NETTemperature.cls*

- V Solution Exploreru jděte do Properties/AssemblyInfo.cs a upravte AssemblyDescription podle obrázku.



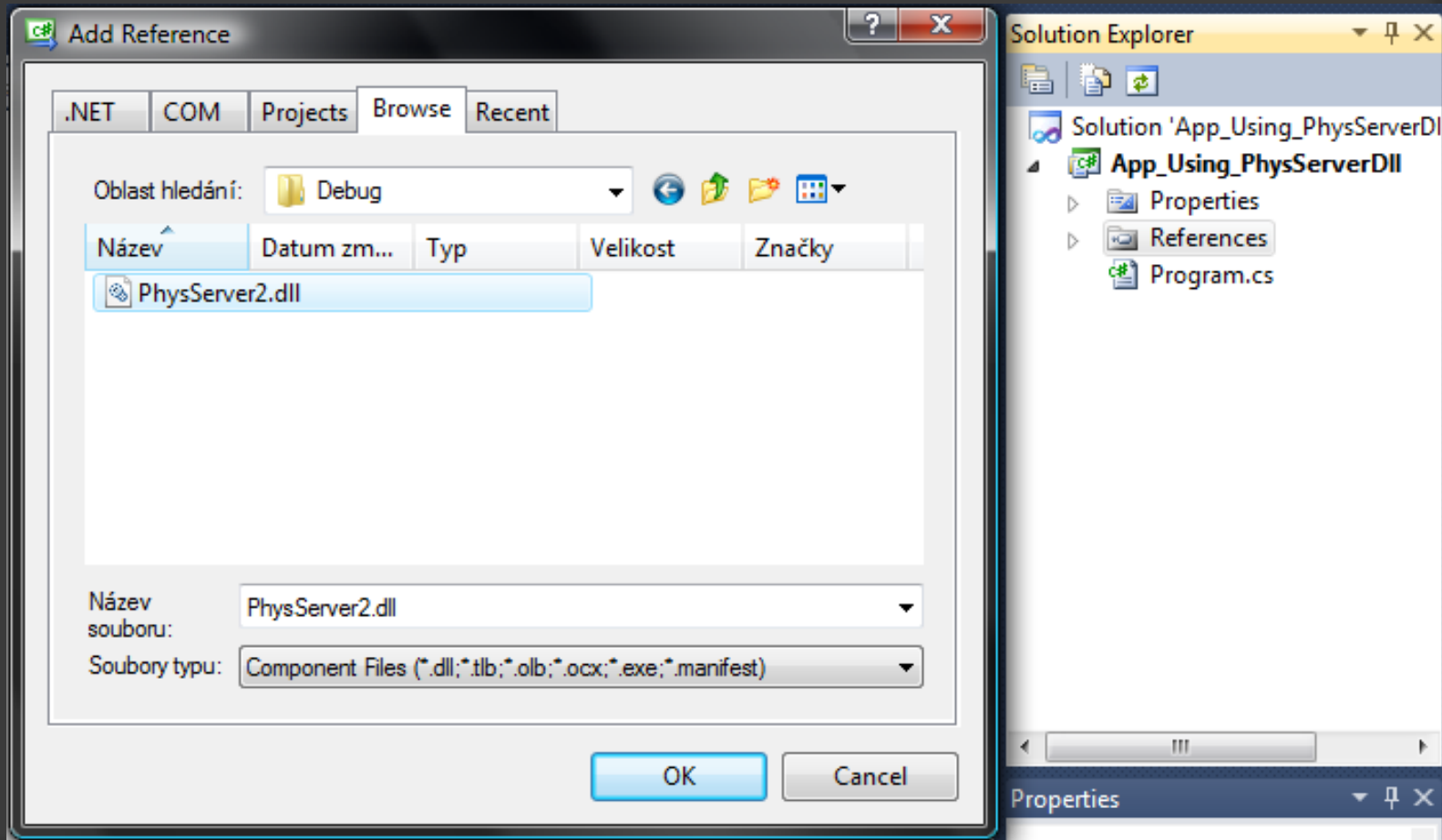
The screenshot shows the Visual Studio IDE with the following components:

- AssemblyInfo.cs* x** (tab):
 - Code content:

```
using System.Reflection;
using System.Runtime.CompilerServices;
using System.Runtime.InteropServices;

// General Information about an assembly is controlled through the
// set of attributes. Change these attribute values to modify the
// associated with an assembly.
[assembly: AssemblyTitle("PhysServer2")]
[assembly: AssemblyDescription(".NET Version of PhysServer")]
[assembly: AssemblyConfiguration("")]
[assembly: AssemblyCompany("")]
[assembly: AssemblyProduct("PhysServer2")]
[assembly: AssemblyCopyright("Copyright © 2011")]
[assembly: AssemblyTrademark("")]
```
- Solution Explorer**:
 - Project: PhysServer2
 - Properties folder expanded, showing **AssemblyInfo.cs**.
 - References folder expanded, showing: Microsoft.CSharp, System, System.Core, System.Data.
- Properties** window:
 - Empty, showing the Properties window for the selected file.

- Vytvořte nový projekt typu ConsoleApp a nazvěte ho **App_usingPhysServerDll**
- Přidejte referenci na vytvořenou dll knihovnu





```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using PhysServer2;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            //vytvořím instanci objektu interfacu definovaného v dll knihovně
            PhysServer2.iTemperature myTemp = new PhysServer2.NETTemperature();
            Console.WriteLine("Zadejte teplotu ve stupních celsia");

            myTemp.Celsius = Convert.ToDouble (Console.ReadLine());
            Console.WriteLine(myTemp.GetFahrenheit().ToString() );
            Console.ReadLine();
        }
    }
}
```

Konec Step by Step Demo aplikace

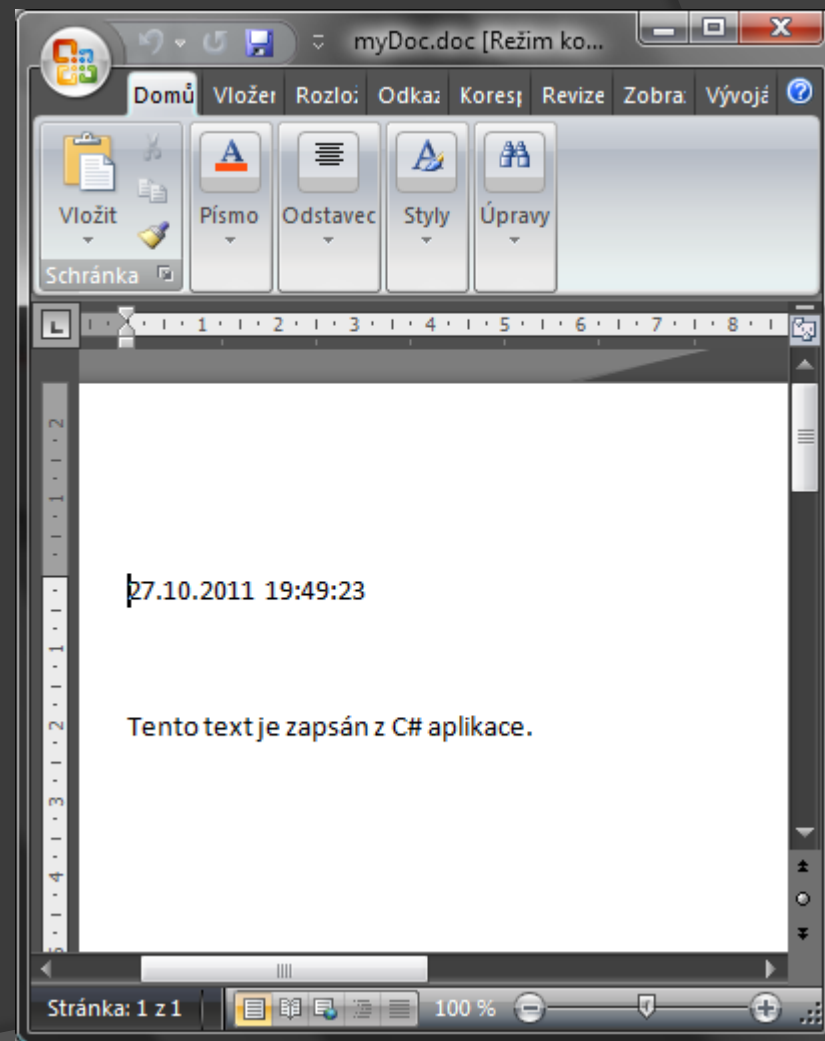


- Tip:
 - Zkuste si dát **breakpoint** na první řádek aplikace a krokovat její průběh – co můžete říci o ladění aplikace a dll knihovny ?
- Řešení v **COM.rar**

Úkol



- Vytvořte aplikaci, která v aplikaci MSWord otevře prázdný předvytvořený dokument mydoc.doc
- Do tohoto dokumentu provede zápis a čas tohoto zápisu
- Aplikaci pak zavře



Pomoc



- ◉ Do projektu musíte přidat referenci na COM aplikace MSWord
- ◉ Budete potřebovat objekty Application, Document a Selection z :
using Microsoft.Office.Interop.Word;
- ◉ Řešení naleznete v
MS_Word_from_Csharp_Demo_01
- ◉ Pokud budete pracovat s řešeným příkladem, pozměňte a rozšiřte jeho funkčnost například vytvořte Aplikaci kde do textboxů zadáte nějaké údaje a po stisku tlačítka se automaticky z těchto údajů vytvoří např. smlouva.

Co byste měli vědět:

- V systému Windows jsou **COM** třídy vyhledávány pomocí tzv. **GUID** (Globally unique identifier) v systémové **DB Registry**,

HKEY_CLASSES_ROOT\CLSID

- Informace o přístupu k **interface** (rozhraní) jsou ve větvi

HKEY_CLASSES_ROOT\Interface

- http://en.wikipedia.org/wiki/Component_Object_Model