

Objektové programování I (C#) – 6

RNDr. Michal Pobucký

michal.pobucky@fpf.slu.cz

Ústav informatiky – zima 2020



Práce se soubory

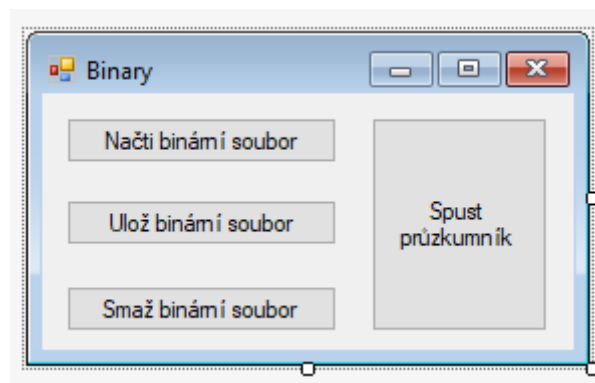
- Pokračujeme v psaní naší aplikace
- Přidejte nový formulář Binary.cs
- A následující ovládací prvky:

Práce se soubory

but_NactiBin

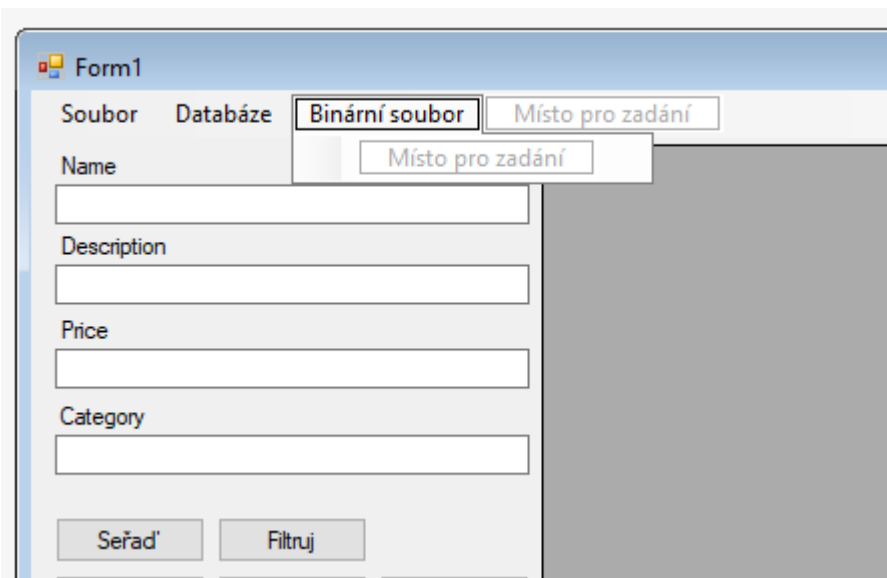
but_UlozBin

but_SmazBin



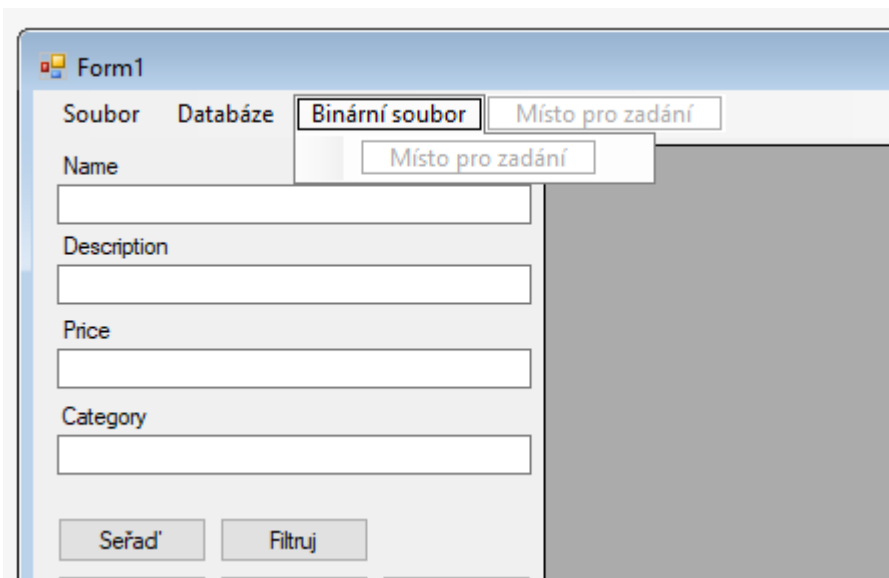
but_Pruzkumnik

Práce se soubory



The image shows a screenshot of a Windows application window titled "Form1". The window has a menu bar with "Soubor" and "Databáze". Below the menu bar, there are two tabs: "Binární soubor" (selected) and "Místo pro zadání". The main area of the form contains several input fields: "Name", "Description", "Price", and "Category". Each field has a corresponding "Místo pro zadání" (input area) label above it. At the bottom of the form, there are two buttons: "Seřad'" and "Filtruj".

Práce se soubory



Vytvoříme událost

```
Počet odkazů: 1
private void BinarniSouborToolStripMenuItem_Click(object sender, EventArgs e)
{
    Binary bin = new Binary();
    bin.ShowDialog();
}
```

Spustit

Práce se soubory

- Přidáme průzkumník souborů

```
Počet odkazů: 4
public partial class Binary : Form
{
    Počet odkazů: 1
    public Binary()
    {
        InitializeComponent();
    }

    Počet odkazů: 1
    private void btn_Průzkumník_Click(object sender, EventArgs e)
    {
        ProcessStartInfo psi = new ProcessStartInfo();
        psi.FileName = @"c:\windows\system32\explorer.exe";
        psi.Arguments = @"c:\\";
        Process.Start(psi);
    }
}
```

Spustit

Práce se soubory

The screenshot displays a C# application interface with a data table, a 'Binary' dialog box, a file explorer window, and a code editor.

Data Table:

Name	ProductID
	1

Binary Dialog Box:

- Načti binární soubor
- Ulož binární soubor
- Smaž binární soubor
- Spust průzkumník

File Explorer (OS (C:)):

Název	Datum
__ZALOHA__	16.07.2
ATEAS Security	20.11.2
eSupport	26.03.2
GINIS	17.05.2
HP Universal Print Driver	20.02.2
LJP1100_P1560_P1600_Full_Solution	07.10.2
My Video	06.03.2
ORACLE	28.07.2
PerfLogs	15.05.2
Program Files	21.10.2
Program Files (x86)	16.10.2
RVERAN	15.10.2
Users	13.09.2
Windows	24.10.2
Windows10Upgrade	21.12.2
Profwiz.config	11.07.2
profwiz.exe	10.07.2
Recovery.txt	28.07.2

Code Editor:

```
1 private void but_Pruzumnik_Click(object sender, EventArgs e)
2 {
3     ProcessStartInfo psi = new ProcessStartInfo
4     {
5         FileName = @"c:\windows\system32\cmd.exe",
6         Arguments = @"c:\\";
7     };
8     Process.Start(psi);
9 }
```

Práce se soubory

- Jde to i jednodušeji

Práce se soubory

- Jde to i jednodušeji

```
Počet odkazů: 1
private void button1_Click(object sender, EventArgs e)
{
    //ProcessStartInfo psi = new ProcessStartInfo();
    //psi.FileName = @"c:\windows\system32\explorer.exe";
    //psi.Arguments = @"c:\\";
    Process.Start(@"c:\");
}
```

Práce se soubory

Počet odkazů: 1

```
private void but_UlozBin_Click(object sender, EventArgs e)
{
    FileStream fs = new FileStream(@"c:\soubor.bin", FileMode.Create);
    BinaryWriter bw = new BinaryWriter(fs);

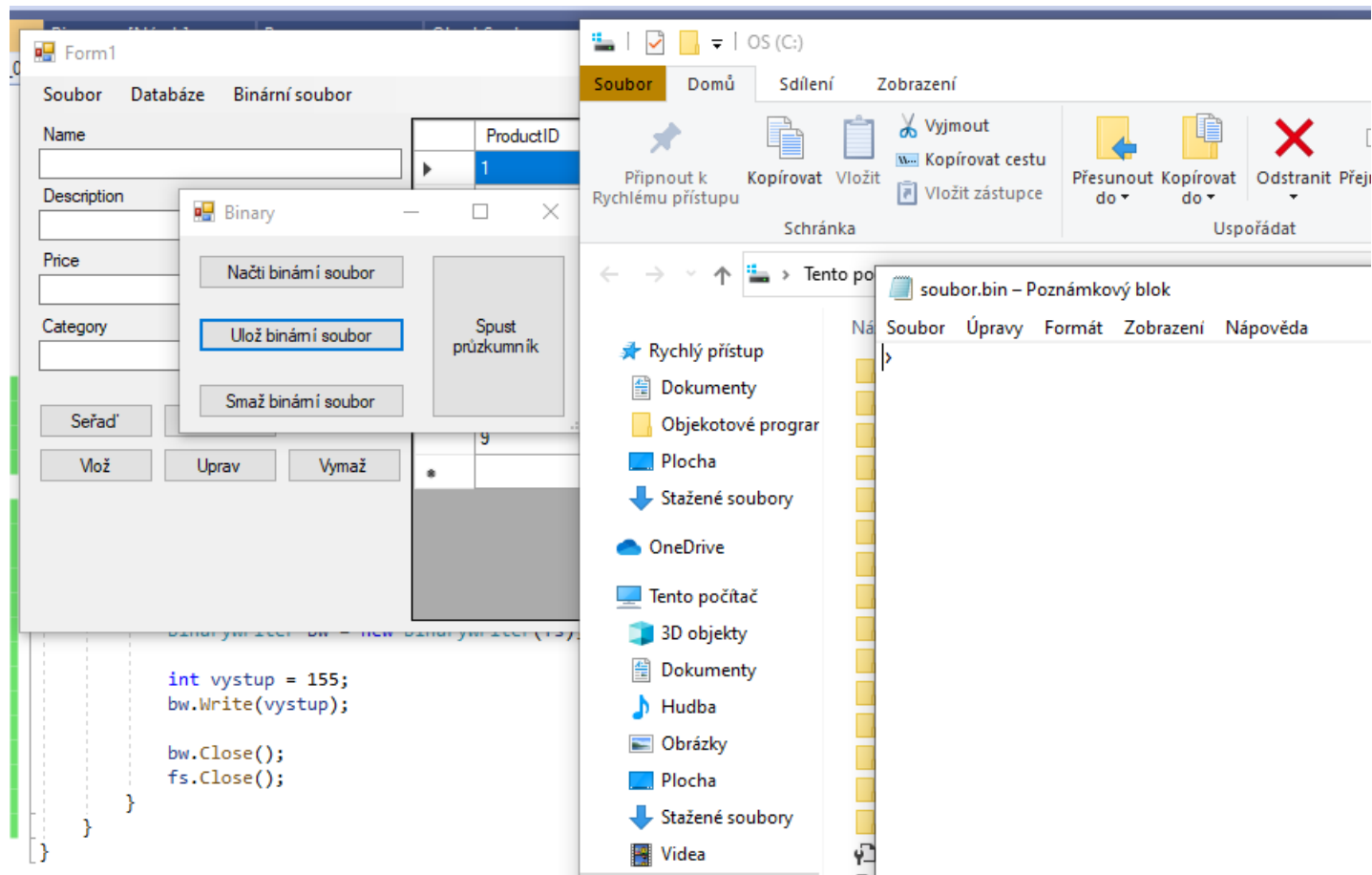
    int vystup = 155;
    bw.Write(vystup);

    bw.Close();
    fs.Close();
}
```

Práce se soubory

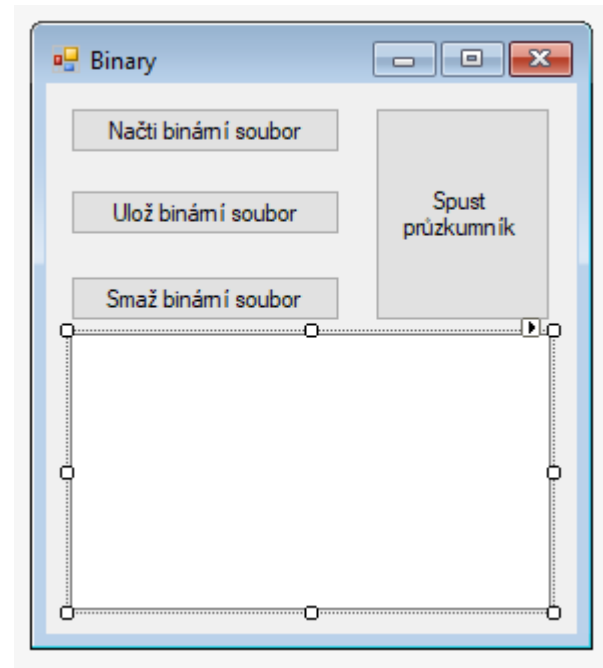
- Spustit
 - Otevřít průzkumník
 - Kliknout na Uložit binární soubor
 - Prohlédnout v průzkumníku obsah souboru bin

Práce se soubory



Práce se soubory

- Přidáme tb_Vystup (nezapomeňte na Multiline)



Práce se soubory

Počet odkazů: 1

```
private void button1_Click(object sender, EventArgs e)
{
    FileStream fs = new FileStream(@"c:\soubor.bin", FileMode.Open);
    BinaryReader br = new BinaryReader(fs);

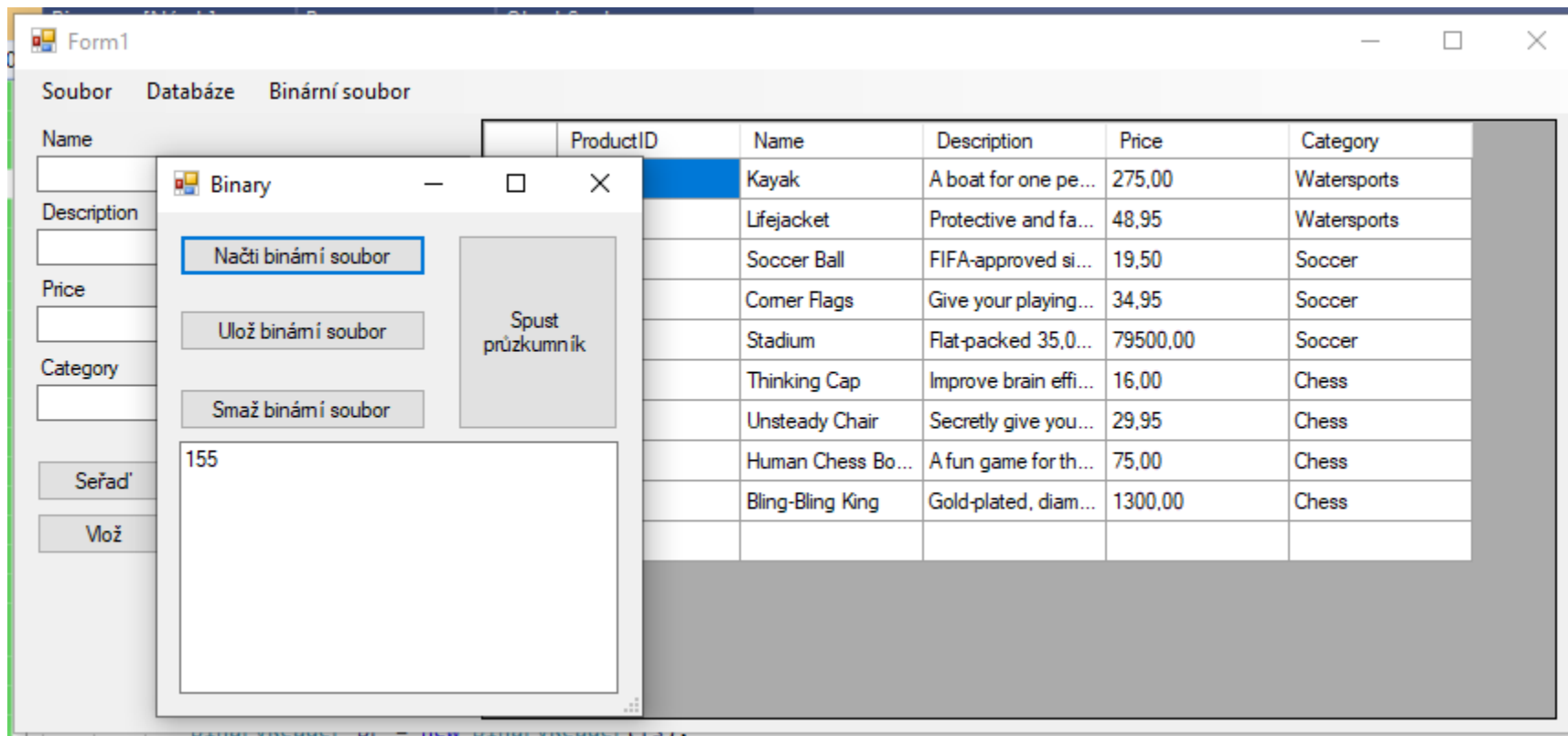
    int vystup = br.ReadInt32();
    tb_Vystup.Text = vystup.ToString();

    br.Close();
    fs.Close();
}
```

Práce se soubory

- Spustit
 - Kliknout na Načti binární soubor

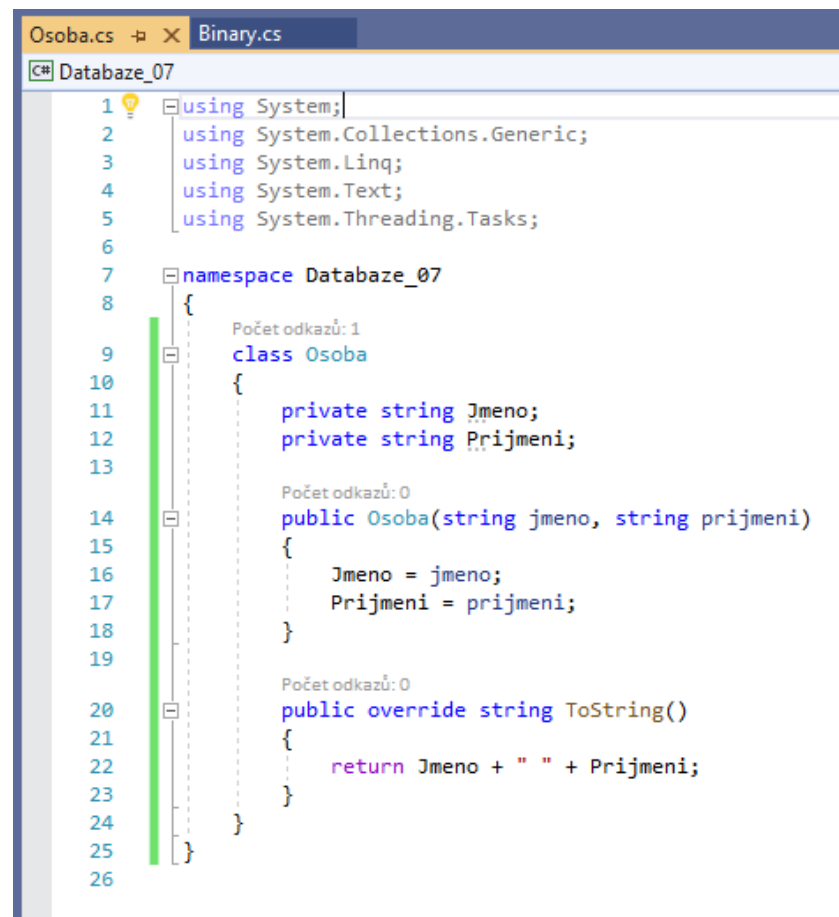
Práce se soubory



Práce se soubory

- BinaryReader a BinaryWriter lze nahradit StreamReader a StreamWriter, pak bude obsah souboru odpovídat textovým řetězcům, které do něj ukládáme (s tím jsme se setkali již dříve)

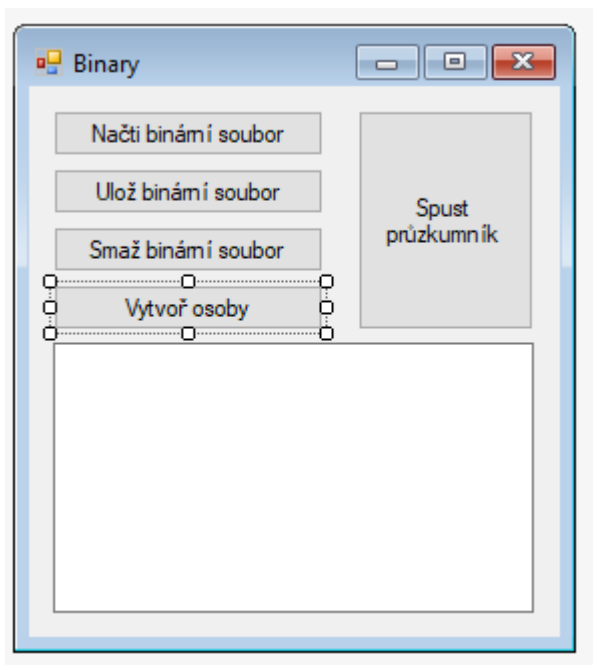
Práce se soubory – serializace



```
Osoba.cs  Binary.cs
C# Database_07
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace Database_07
8 {
9     class Osoba
10    {
11        private string Jmeno;
12        private string Prijmeni;
13
14        public Osoba(string jmeno, string prijmeni)
15        {
16            Jmeno = jmeno;
17            Prijmeni = prijmeni;
18        }
19
20        public override string ToString()
21        {
22            return Jmeno + " " + Prijmeni;
23        }
24    }
25 }
26
```

Vytvoříme novou třídu Osoba.cs

Práce se soubory – serializace



but_VytvorOsoby

Práce se soubory – serializace

```
Binary.cs [Návrh] | Binary.cs [X]
C# Database_07 | Database_07.Binary
1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Diagnostics;
6  using System.Drawing;
7  using System.IO;
8  using System.Linq;
9  using System.Text;
10 using System.Threading.Tasks;
11 using System.Windows.Forms;
12 using System.Runtime.Serialization.Formatters.Binary;
13
14 namespace Database_07
15 {
16     Počet odkazů: 4
17     public partial class Binary : Form
18     {
19         List<Osoba> moje = new List<Osoba>();
20         string[] Jmena = { "Karel", "Josef", "Adam", "Jan", "Michal" };
21         string[] Prijmeni = { "Nový", "Starý", "Malý", "Velký", "Chytrý" };
22
23         Počet odkazů: 1
24         public Binary()
25         {
26             InitializeComponent();
27         }
28     }
29 }
```

Práce se soubory – serializace

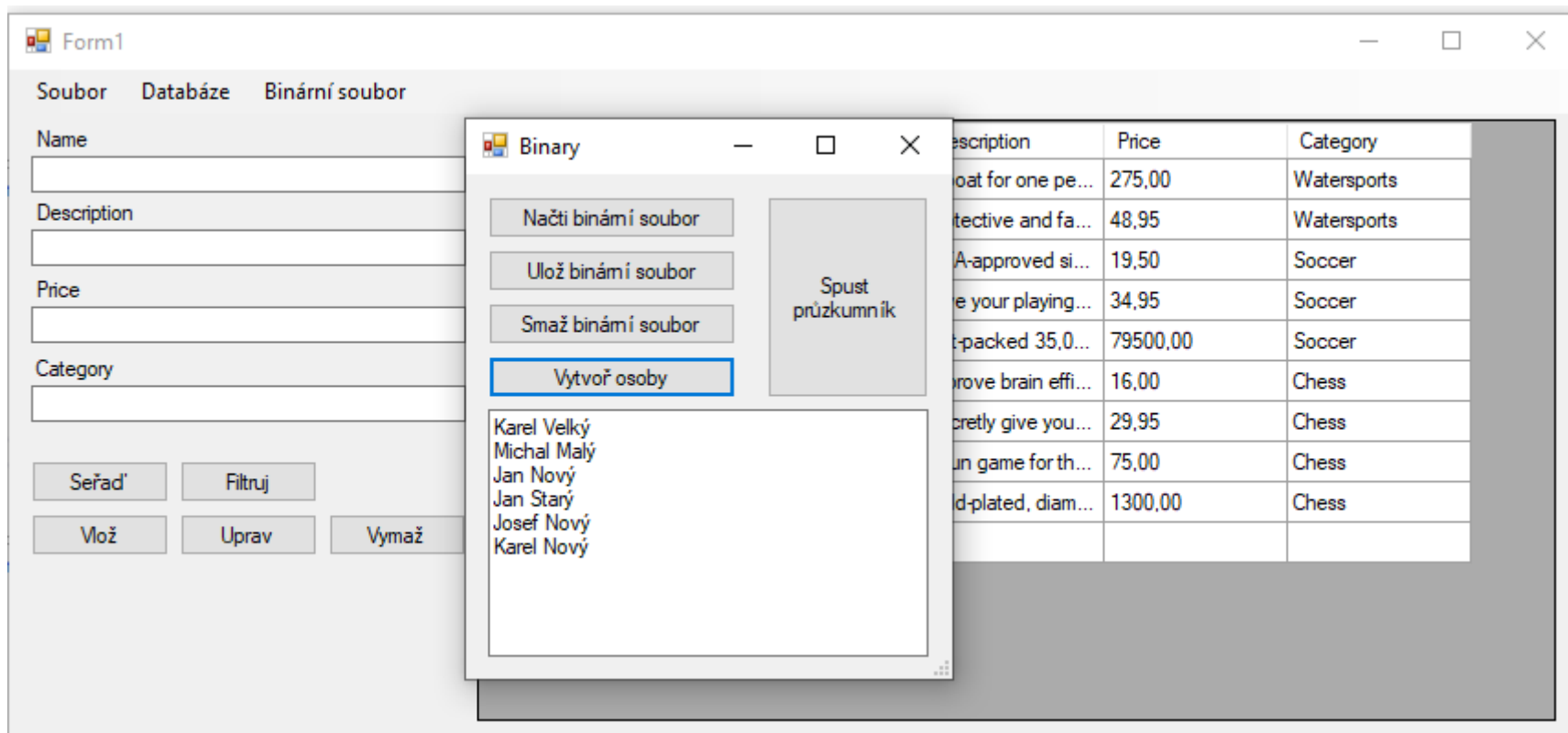
Počet odkazů: 1

```
private void but_VytvorOsoby_Click(object sender, EventArgs e)
{
    Random rand = new Random();
    moje.Add(new Osoba(Jmena[rand.Next(Jmena.Length)], Prijmeni[rand.Next(Prijmeni.Length)]));
    tb_Vystup.Text = "";
    foreach (Osoba o in moje)
    {
        tb_Vystup.Text += o.ToString() + Environment.NewLine;
    }
}
```

Práce se soubory – serializace

- Spustit
 - Vytvářejte osoby

Práce se soubory – serializace



Práce se soubory – serializace

```
namespace Database_07
{
    [Serializable]
    Počet odkazů: 5
    class Osoba
    {
        private string Jmeno;
        private string Prijmeni;

        Počet odkazů: 1
        public Osoba(string jmeno, string prijmeni)
        {
            Jmeno = jmeno;
            Prijmeni = prijmeni;
        }

        Počet odkazů: 1
        public override string ToString()
        {
            return Jmeno + " " + Prijmeni;
        }
    }
}
```


Práce se soubory – serializace

Počet odkazů: 1

```
private void btn_UlozBin_Click(object sender, EventArgs e)
{
    FileStream fs = new FileStream(@"c:\soubor.bin", FileMode.Create);
    //BinaryWriter bw = new BinaryWriter(fs);

    //int vystup = 155;
    //bw.Write(vystup);

    BinaryFormatter bf = new BinaryFormatter();
    bf.Serialize(fs, moje);

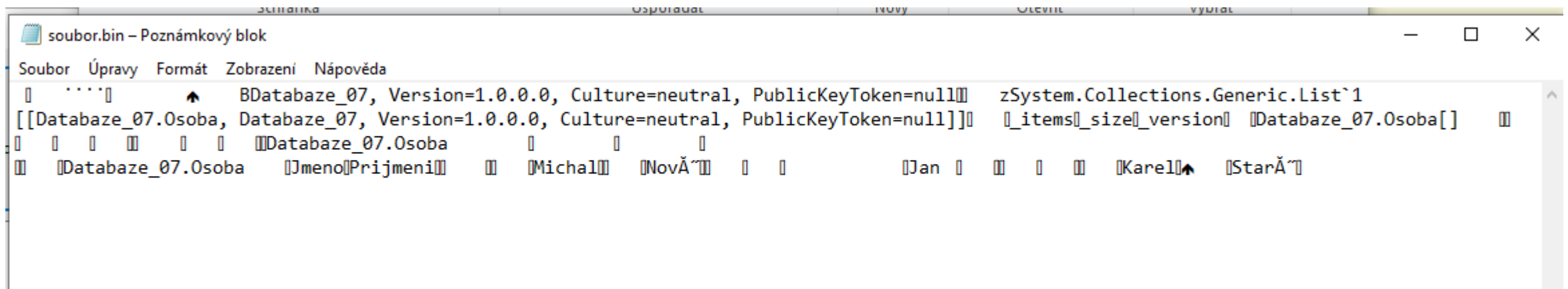
    //bw.Close();
    fs.Close();
}
```

Práce se soubory – serializace

- Spustit
 - Vytvořit osoby
 - Uložit binární soubor
 - Otevřít průzkumníka
 - Prohlédnout obsah souboru BIN

Práce se soubory – serializace

- Spustit
 - Vytvořit osoby
 - Uložit binární soubor
 - Otevřít průzkumníka
 - Prohlédnout obsah souboru BIN



```
soubor.bin - Poznámkový blok
Soubor  Úpravy  Formát  Zobrazení  Nápověda
[[{"Database_07.Osoba", "Database_07", "Version=1.0.0.0", "Culture=neutral", "PublicKeyToken=null"}], [{"_items_size": 1, "_version": "Database_07.Osoba[]"}]]
[[{"Database_07.Osoba", "Jmeno", "Prijmeni", "Michal", "Nováček", "Jan", "Karel", "Staráček"}]]
```

Práce se soubory – serializace

- Refaktorujeme obsah zobrazení z `but_VytvorOsoby_Click` na metodu `ZobrazMoje()`

Práce se soubory – serializace

- Refaktorujeme obsah zobrazení z `but_VytvorOsoby_Click` na metodu `ZobrazMoje()`

```
Počet odkazů: 1
private void but_VytvorOsoby_Click(object sender, EventArgs e)
{
    Random rand = new Random();
    moje.Add(new Osoba(Jmena[rand.Next(Jmena.Length)], Prijmeni[rand.Next(Prijmeni.Length)]));
    ZobrazMoje();
}

Počet odkazů: 1
private void ZobrazMoje()
{
    tb_Vystup.Text = "";
    foreach (Osoba o in moje)
    {
        tb_Vystup.Text += o.ToString() + Environment.NewLine;
    }
}
```

Práce se soubory – serializace

Počet odkazů: 1

```
private void button1_Click(object sender, EventArgs e)
{
    FileStream fs = new FileStream(@"c:\soubor.bin", FileMode.Open);
    //BinaryReader br = new BinaryReader(fs);

    //int vystup = br.ReadInt32();
    //tb_Vystup.Text = vystup.ToString();

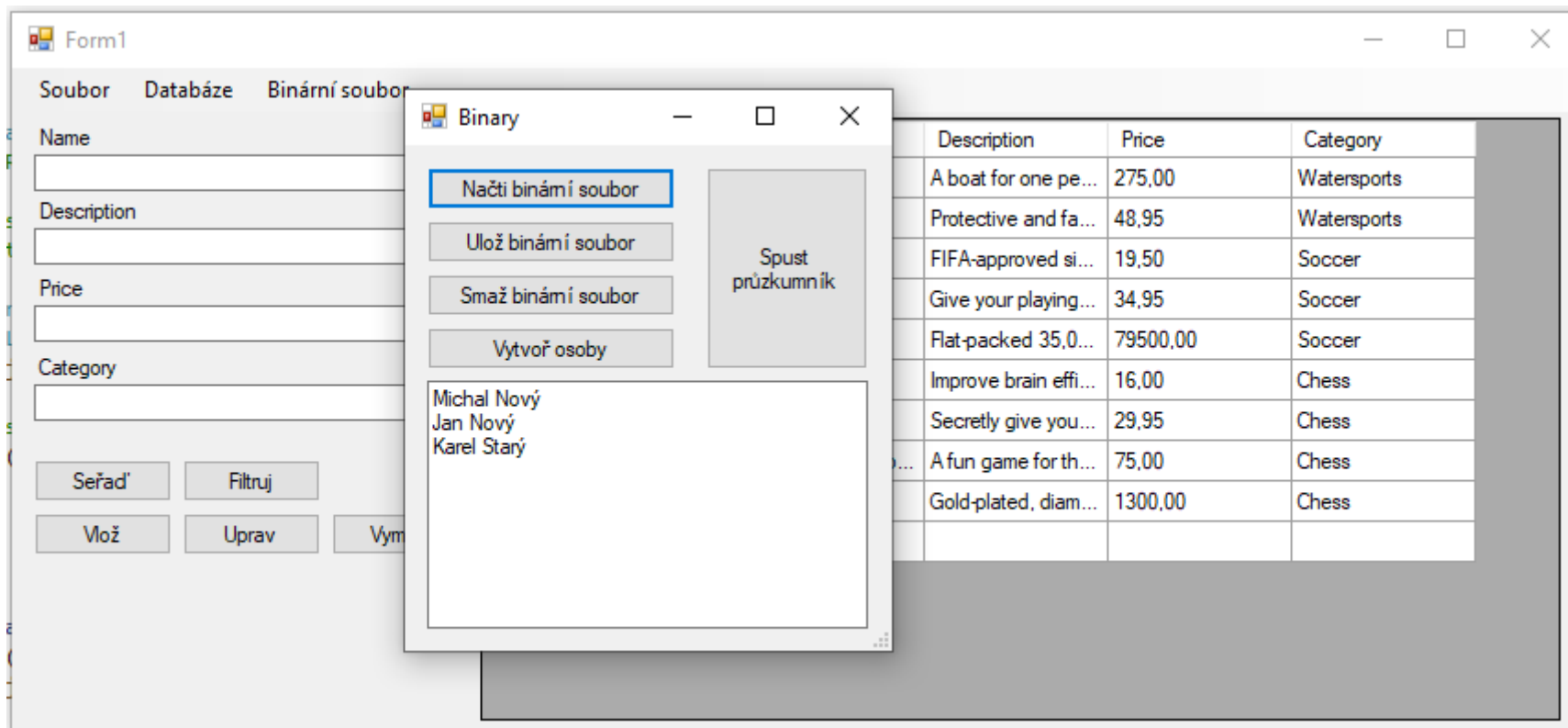
    BinaryFormatter bf = new BinaryFormatter();
    moje = (List<Osoba>)bf.Deserialize(fs);
    ZobrazMoje();

    //br.Close();
    fs.Close();
}
```

Práce se soubory – serializace

- Spustit
 - Načíst binární soubor

Práce se soubory – serializace



Práce se soubory – serializace

- Vytvoříme událost pro smazání binárního souboru

Práce se soubory – serializace

```
Počet odkazů: 4
public partial class Binary : Form
{
    List<Osoba> moje = new List<Osoba>();
    string[] Jmena = { "Karel", "Josef", "Adam", "Jan", "Michal" };
    string[] Prijmeni = { "Nový", "Starý", "Malý", "Velký", "Chytrý" };
    string soubor = @"c:\soubor.bin";

    Počet odkazů: 1
    public Binary()
    {
        InitializeComponent();
    }

    Počet odkazů: 1
    private void but_Pruzkumnik_Click(object sender, EventArgs e) { ... }

    Počet odkazů: 1
    private void but_UlozBin_Click(object sender, EventArgs e)
    {
        FileStream fs = new FileStream(soubor, FileMode.Create);
        //BinaryWriter bw = new BinaryWriter(fs);

        //int vystup = 155;
        //bw.Write(vystup);

        BinaryFormatter bf = new BinaryFormatter();
        bf.Serialize(fs, moje);

        //bw.Close();
        fs.Close();
    }
}
```

Nejprve vytvoříme
globální proměnnou

Práce se soubory – serializace

```
Počet odkazů: 4
public partial class Binary : Form
{
    List<Osoba> moje = new List<Osoba>();
    string[] Jmena = { "Karel", "Josef", "Adam", "Jan", "Michal" };
    string[] Prijmeni = { "Nový", "Starý", "Malý", "Velký", "Chytrý" };
    string soubor = @"c:\soubor.bin";

    Počet odkazů: 1
    public Binary()
    {
        InitializeComponent();
    }

    Počet odkazů: 1
    private void but_Pruzkumnik_Click(object sender, EventArgs e)
    {
        Počet odkazů: 1
        private void but_UlozBin_Click(object sender, EventArgs e)
        {
            FileStream fs = new FileStream(soubor, FileMode.Create);
            //BinaryWriter bw = new BinaryWriter(fs);

            //int vystup = 155;
            //bw.Write(vystup);

            BinaryFormatter bf = new BinaryFormatter();
            bf.Serialize(fs, moje);

            //bw.Close();
            fs.Close();
        }
    }
}
```

Nejprve vytvoříme
globální proměnnou

```
Počet odkazů: 1
private void but_NactiBin_Click(object sender, EventArgs e)
{
    FileStream fs = new FileStream(soubor, FileMode.Open);
    //BinaryReader br = new BinaryReader(fs);

    //int vystup = br.ReadInt32();
    //tb_Vystup.Text = vystup.ToString();

    BinaryFormatter bf = new BinaryFormatter();
    moje = (List<Osoba>)bf.Deserialize(fs);
    ZobrazMoje();

    //br.Close();
    fs.Close();
}
```

Práce se soubory – serializace

```
Počet odkazů: 1
private void btn_SmazBin_Click(object sender, EventArgs e)
{
    File.Delete(soubor);
    moje.Clear();
    ZobrazMoje();
}
```

Práce se soubory – serializace

- Spustit
 - Načíst binární soubor
 - Smazat binární soubor
 - Načíst binární soubor

Co se stane?

Práce se soubory – serializace

- Spustit
 - Načíst binární soubor
 - Smazat binární soubor
 - Načíst binární soubor

Co se stane?

```
Počet odkazů: 1
private void but_NactiBin_Click(object sender, EventArgs e)
{
    FileStream fs = new FileStream(soubor, FileMode.Open);
    //BinaryReader br = new BinaryReader(fs);

    //int vystup = br.ReadInt32();
    //tb_Vystup.Text = vystup.ToString();

    BinaryFormatter bf = new BinaryFormatter();
    moje = (List<Osoba>)bf.Deserialize(fs);
    ZobrazMoje();

    //br.Close();
    fs.Close();
}

Počet odkazů: 1
private void but_VytvorOsoby_Click(object sender, EventArgs e)
{
    Random rand = new Random();
    //...
}
```

Neošetřená výjimka

System.IO.FileNotFoundException: Soubor c:\soubor.bin nebyl

Tato výjimka se původně vyvolala na tomto zásobníku volání:

- [Externí kód]
- Database_07.Binary.but_NactiBin_Click(object, System.EventArgs) v Bi
- [Externí kód]
- Database_07.Form1.binarniSouborToolStripMenuItem.Click(object, Sv

[Zobrazit podrobnosti](#) | [Kopírovat podrobnosti](#) | [Spustit relaci Live Share](#)

▲ Nastavení výjimek

- Přerušit při vyvolání výjimky tohoto typu
- Kromě případů, kdy je vyvolána z:
 - Database_07.exe

Práce se soubory – serializace

- Ošetříme výjimku

Počet odkazů: 1

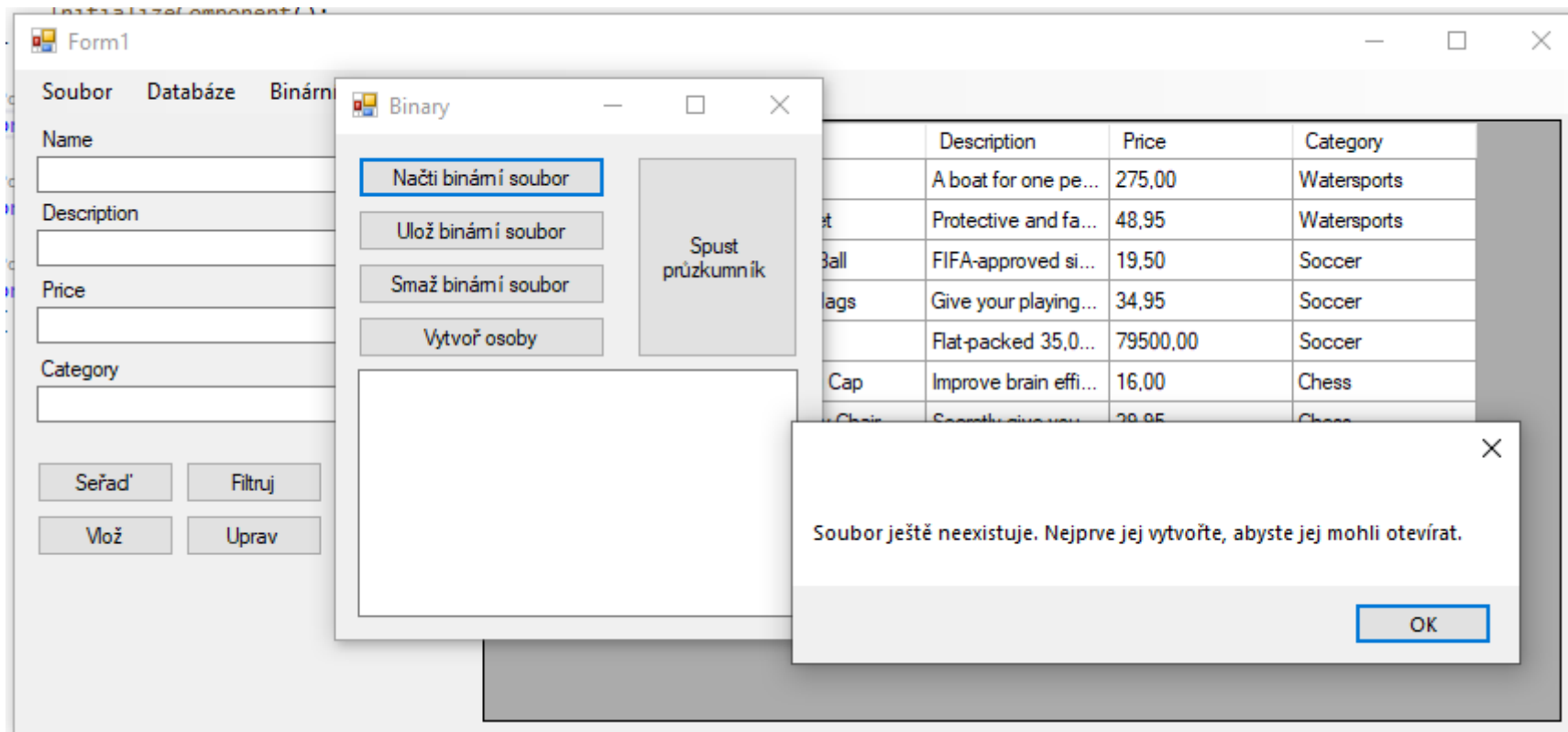
```
private void but_NactiBin_Click(object sender, EventArgs e)
{
    try
    {
        FileStream fs = new FileStream(soubor, FileMode.Open);
        //BinaryReader br = new BinaryReader(fs);

        //int vystup = br.ReadInt32();
        //tb_Vystup.Text = vystup.ToString();

        BinaryFormatter bf = new BinaryFormatter();
        moje = (List<Osoba>)bf.Deserialize(fs);
        ZobrazMoje();

        //br.Close();
        fs.Close();
    }
    catch (FileNotFoundException)
    {
        MessageBox.Show("Soubor ještě neexistuje. Nejprve jej vytvořte, abyste jej mohli otevírat.");
    }
}
```

Práce se soubory – serializace

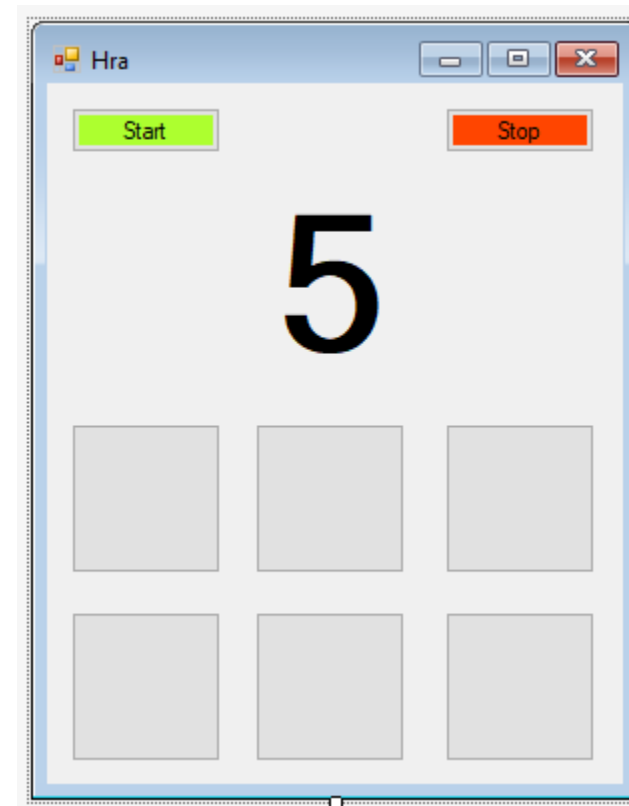


Hra

- Vytvoříme jednoduchou hru
- Naučíme se používat časovač - Timer

Hra

- Vytvořte nový formulář – Hra.cs
 - but_Start, BackColor = GreenYellow
 - but_Stop, BackColor = OrangeRed
 - lab_Time, Font = Microsoft Sans Serif, 72pt; Bold
Text = 5
 - but_1 až but_6, Size=75; 75, Text=„“
- v grafickém editoru přidáme ještě Timer timer1
 - Interval = 1000



Hra

- Obsloužíme událost Timeru Tick

```
namespace Databaze_07
{
    Počet odkazů: 2
    public partial class Hra : Form
    {
        int cas = 5;

        Počet odkazů: 0
        public Hra()
        {
            InitializeComponent();
        }

        Počet odkazů: 1
        private void timer1_Tick(object sender, EventArgs e)
        {
            cas--;
            lab_Time.Text = cas.ToString();
        }
    }
}
```

Hra

- Vytvoříme události buttonu Start a Stop

Počet odkazů: 1

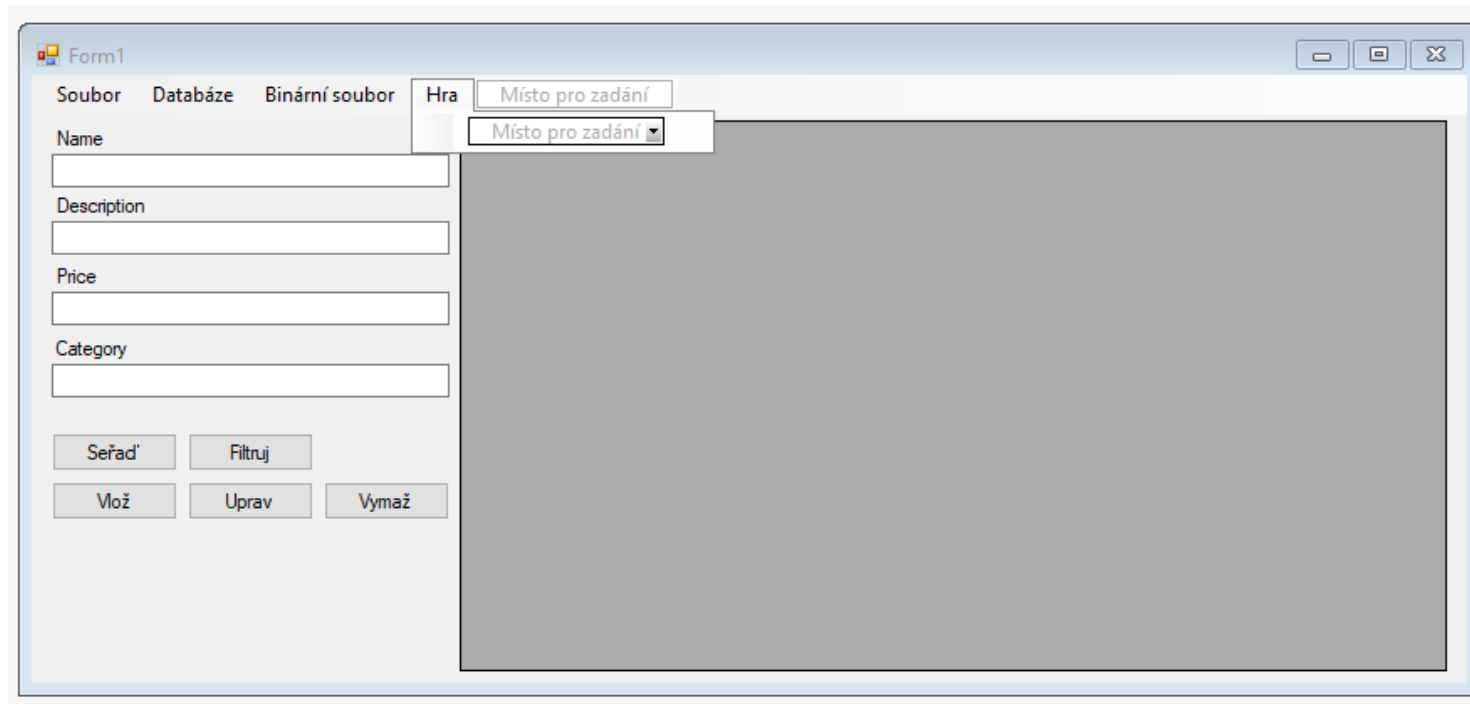
```
private void but_Start_Click(object sender, EventArgs e)
{
    timer1.Start();
}
```

Počet odkazů: 1

```
private void but_Stop_Click(object sender, EventArgs e)
{
    timer1.Stop();
}
```

Hra

- Vytvoříme odkaz v menu



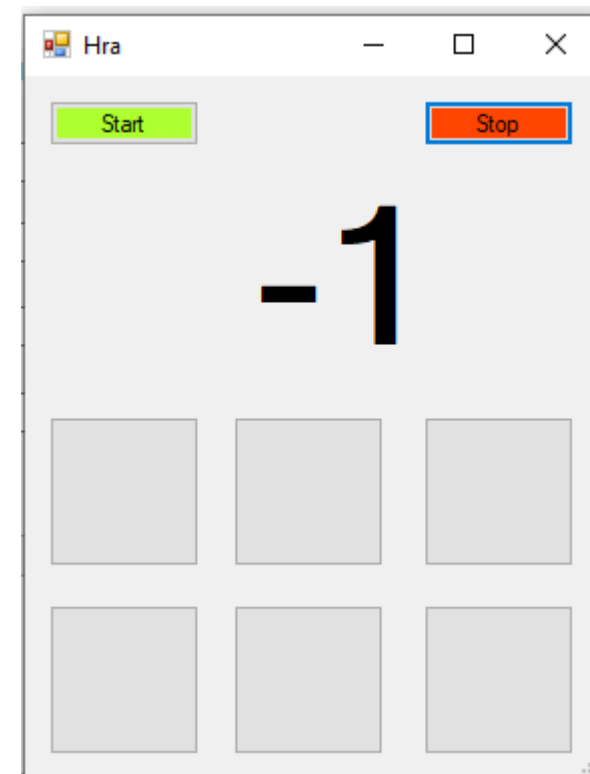
Hra

- Obsloužíme odkaz v menu

```
Počet odkazů: 1
private void hraToolStripMenuItem_Click(object sender, EventArgs e)
{
    Hra hra = new Hra();
    hra.ShowDialog();
}
```

Spustit

Hra



Hra

- Na šesti buttonech se zobrazí šest čísel. Jedno se bude lišit v paritě (buď bude sudé a ostatní lichá anebo naopak). Uživatel musí kliknout na odlišné číslo. Pokud se splete – prohrál. Pokud mu vyprší čas – prohrál.

Hra

- Nastavení čísel

Počet odkazů: 1

```
private void NastavHodnoty()
{
    Random rand = new Random();
    int parita = rand.Next(2);
    int pozice = rand.Next(6);

    List<int> seznam = new List<int>();
    for (int i = 0; i < 6; i++)
    {
        int hodnota = rand.Next(50);
        hodnota *= 2; // všechno sudá čísla

        if (parita == 1) // všechno lichá čísla
        {
            hodnota++;
        }

        seznam.Add(hodnota);
    }

    seznam[pozice]--; // vytvříme odlišné

    for (int i = 1; i < 7; i++)
    {
        this.Controls.Find("but_" + i, true).FirstOrDefault().Text = seznam[i-1].ToString();
    }
}
```

Hra

- Nastavení čísel

```
Počet odkazů: 1
private void but_Start_Click(object sender, EventArgs e)
{
    timer1.Start();
    NastavHodnoty();
}
```

Spustit

```
Počet odkazů: 1
private void NastavHodnoty()
{
    Random rand = new Random();
    int parita = rand.Next(2);
    int pozice = rand.Next(6);

    List<int> seznam = new List<int>();
    for (int i = 0; i < 6; i++)
    {
        int hodnota = rand.Next(50);
        hodnota *= 2; // všechno sudá čísla

        if (parita == 1) // všechno lichá čísla
        {
            hodnota++;
        }

        seznam.Add(hodnota);
    }

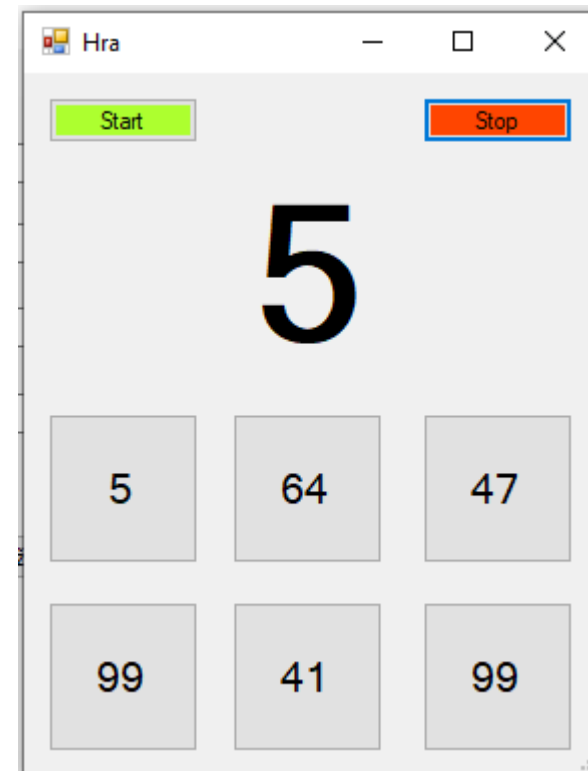
    seznam[pozice]--; // vytvoříme odlišné

    for (int i = 1; i < 7; i++)
    {
        this.Controls.Find("but_" + i, true).FirstOrDefault().Text = seznam[i-1].ToString();
    }
}
```

Hra



Nastavíme velikost
fontu u buttonů na 16



Hra

- Vytvoříme globální proměnnou

```
namespace Database_07
{
    Počet odkazů: 4
    public partial class Hra : Form
    {
        int cas = 5;
        int odlisna;
        Počet odkazů: 1
        public Hra()
        {
            InitializeComponent();
        }
    }
}
```

Hra

- Nastavíme globální proměnnou

Počet odkazů: 2

```
private void NastavHodnoty()
{
    Random rand = new Random();
    int parita = rand.Next(2);
    int pozice = rand.Next(6);

    List<int> seznam = new List<int>();
    for (int i = 0; i < 6; i++)
    {
        int hodnota = rand.Next(50);
        hodnota *= 2; // všechno sudá čísla

        if (parita == 1) // všechno lichá čísla
        {
            hodnota++;
        }

        seznam.Add(hodnota);
    }

    seznam[pozice]--; // vytvoříme odlišné

    for (int i = 1; i < 7; i++)
    {
        this.Controls.Find("but_" + i, true).FirstOrDefault().Text = seznam[i-1].ToString();
    }

    odlišna = pozice + 1;
    cas = 5;
}
```

Hra

Počet odkazů: 2

```
private void KonecHry()  
{  
    timer1.Stop();  
    MessageBox.Show("Konec hry.");  
    this.Close();  
}
```

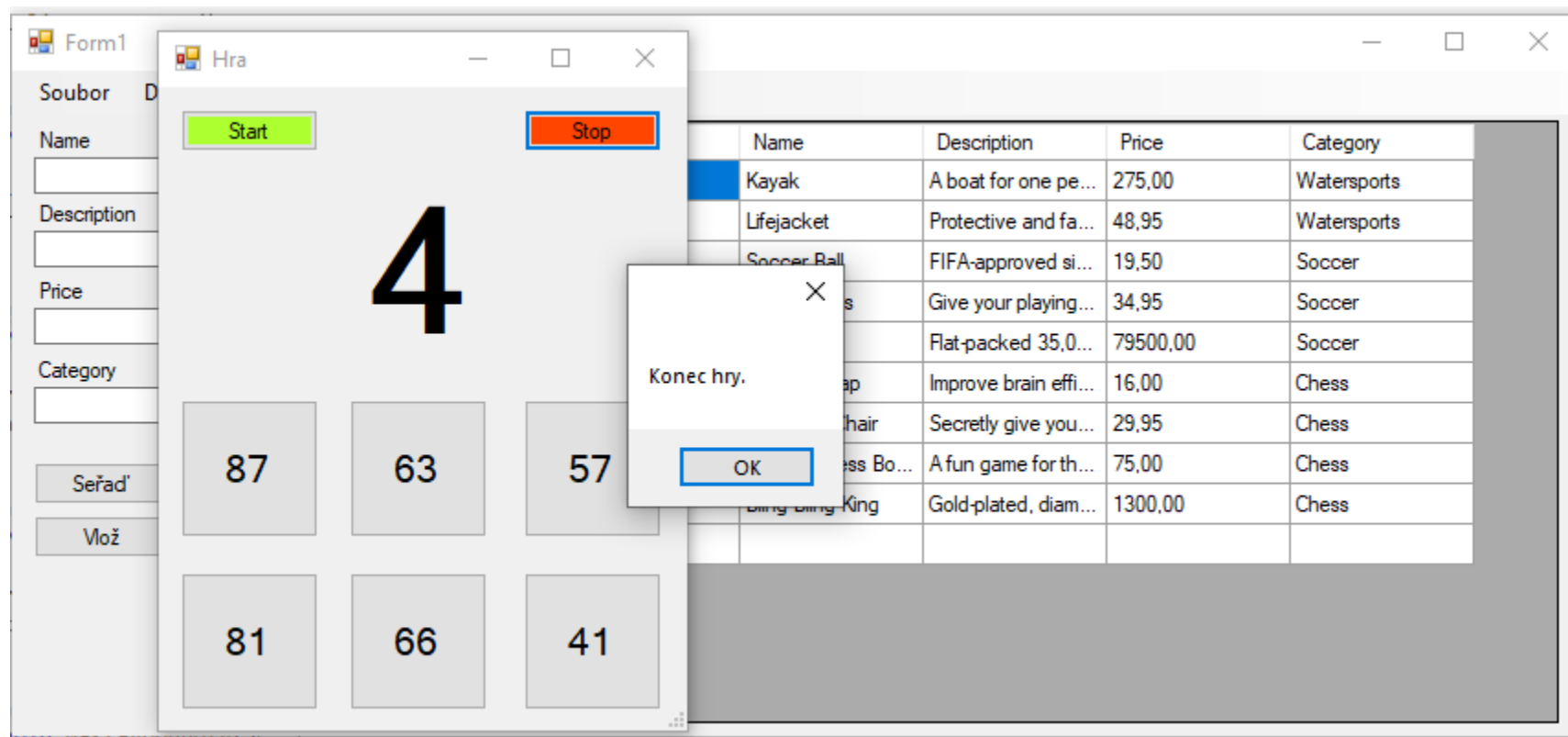
Hra

Počet odkazů: 1

```
private void but_Stop_Click(object sender, EventArgs e)
{
    timer1.Stop();
    KonecHry();
}
```

Spustit

Hra



Hra

- Vytvoříme globální proměnnou

```
namespace Database_07
{
    Počet odkazů: 4
    public partial class Hra : Form
    {
        int cas = 5;
        int odlišna;
        int uspech = 0;

        Počet odkazů: 1
        public Hra(...)

        Počet odkazů: 1
        private void timer1_Tick(object
```

Hra

- Obsloužíme button 1

```
Počet odkazů: 1
private void but_1_Click(object sender, EventArgs e)
{
    if (odlisna == 1)
    {
        uspech++;
        NastavHodnoty();
    }
    else
    {
        KonecHry();
    }
}
```

Hra

- Obsloužíme button 1

```
Počet odkazů: 1
private void but_1_Click(object sender, EventArgs e)
{
    if (odlisna == 1)
    {
        uspech++;
        NastavHodnoty();
    }
    else
    {
        KonecHry();
    }
}
```

Kód bude identický i u dalších buttonů, proto refaktorizujeme =>

Hra

```
Počet odkazů: 1
private void Porovnej(int but)
{
    if (odlisna == but)
    {
        uspech++;
        NastavHodnoty();
    }
    else
    {
        KonecHry();
    }
}

Počet odkazů: 1
private void but_1_Click(object sender, EventArgs e)
{
    Porovnej(1);
}
```

Hra

- Obsloužíme zbylé buttony

```
Počet odkazů: 1  
private void but_1_Click(object sender, EventArgs e)  
{  
    Porovnej(1);  
}
```

```
Počet odkazů: 1  
private void but_2_Click(object sender, EventArgs e)  
{  
    Porovnej(2);  
}
```

```
Počet odkazů: 1  
private void but_3_Click(object sender, EventArgs e)  
{  
    Porovnej(3);  
}
```

```
Počet odkazů: 1  
private void but_4_Click(object sender, EventArgs e)  
{  
    Porovnej(4);  
}
```

```
Počet odkazů: 1  
private void but_5_Click(object sender, EventArgs e)  
{  
    Porovnej(5);  
}
```

```
Počet odkazů: 1  
private void but_6_Click(object sender, EventArgs e)  
{  
    Porovnej(6);  
}
```

Hra

- Odladíme drobnosti
- Nezobrazuje hodnotu 5

```
Počet odkazů: 2
private void NastavHodnoty()
{
    Random rand = new Random();
    int parita = rand.Next(2);
    int pozice = rand.Next(6);

    List<int> seznam = new List<int>();
    for (int i = 0; i < 6; i++)
    {
        int hodnota = rand.Next(50);
        hodnota *= 2; // všechno sudá čísla

        if (parita == 1) // všechno lichá čísla
        {
            hodnota++;
        }

        seznam.Add(hodnota);
    }

    seznam[pozice]--; // vytvříme odlišné

    for (int i = 1; i < 7; i++)
    {
        this.Controls.Find("but_" + i, true).Fir

    }

    odlisna = pozice + 1;
    cas = 5;
    timer1.Stop();
    lab_Time.Text = cas.ToString();
    timer1.Start();
}
```

Hra

- Odladíme drobnosti
- Na nule se nezastaví, pokračuje do mínusu

```
Počet odkazů: 1
private void timer1_Tick(object sender, EventArgs e)
{
    cas--;
    lab_Time.Text = cas.ToString();

    if (cas == 0)
    {
        KonecHry();
    }
}
```

Hra

- Odladíme drobnosti

```
Počet odkazů: 3  
private void KonecHry()  
{  
    timer1.Stop();  
    string zprava = "Konec hry." + Environment.NewLine + "Tvoje skóre: " + uspech;  
    MessageBox.Show(this, zprava, "KONEC", MessageBoxButtons.OK, MessageBoxIcon.Information);  
    this.Close();  
}
```


Hra

The screenshot displays a Windows application with three main windows:

- Form1**: A data entry form with fields for Name, Description, Price, and Category, and buttons for 'Seřad', 'Filtruj', 'Vlož', and 'Uprav'.
- Hra**: A game window featuring a 'Start' button, a 'Stop' button, a large score display showing '4', and a grid of numbers (37, 95, 8, 33, 67, 68). The number '95' is highlighted with a blue border.
- KONEC**: A dialog box with an information icon and the text 'Konec hry. Tvoje skóre: 4', with an 'OK' button.

In the background, a table with the following data is visible:

Description	Price	Category
A boat for one pe...	275,00	Watersports
Protective and fa...	48,95	Watersports
...	19,50	Soccer
...	34,95	Soccer
...	79500,00	Soccer
...	16,00	Chess
...	29,95	Chess
...	75,00	Chess
...	1300,00	Chess

Hra

- Vylepšíme časovač

Hra

- Vylepšíme časovač

```
namespace Database_07
{
    Počet odkazů: 4
    public partial class Hra : Form
    {
        int maxCas = 500;
        int cas;
        int odlisna;
        int uspech = 0;

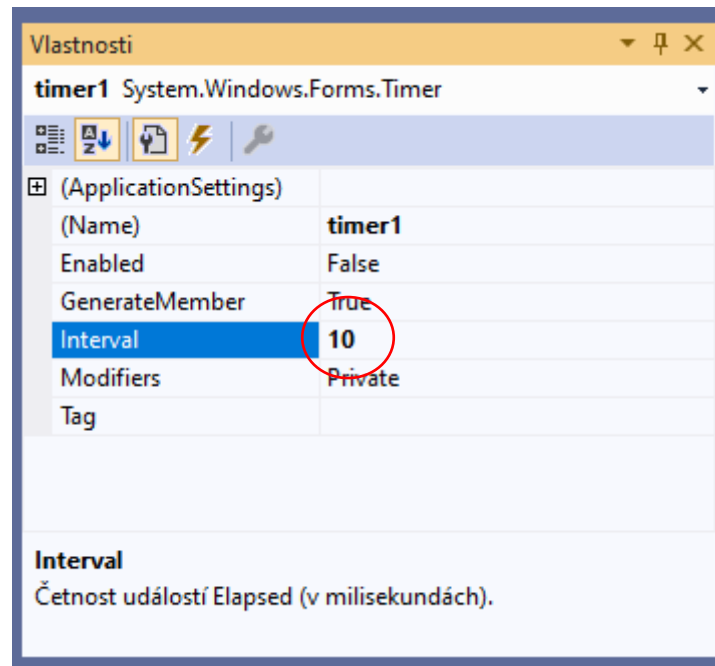
        Počet odkazů: 1
        public Hra()...

        Počet odkazů: 1
        private void timer1_Tick(object ser

        Počet odkazů: 1
        private void but_Start_Click(object
```

Hra

- Vylepšíme časovač



Hra

- Vylepšíme časovač

```
private void NastavHodnoty()
{
    Random rand = new Random();
    int parita = rand.Next(2);
    int pozice = rand.Next(6);

    List<int> seznam = new List<int>();
    for (int i = 0; i < 6; i++)
    {
        int hodnota = rand.Next(50);
        hodnota *= 2; // všechno sudá čísla

        if (parita == 1) // všechno lichá čísla
        {
            hodnota++;
        }

        seznam.Add(hodnota);
    }

    seznam[pozice]--; // vytvoříme odlišné

    for (int i = 1; i < 7; i++)
    {
        this.Controls.Find("but_" + i, true).FirstOrDefault
    }

    odlisna = pozice + 1;
    cas = maxCas;
    timer1.Stop();
    lab_Time.Text = cas.ToString();
    timer1.Start();
}
```

Hra

- Vylepšíme časovač

Upravíme label

