

# Típus, változó

## Fogalmak

*Változó:* Egy memóriaszelet ahol adatot tárolunk, névvel azonosítjuk

*Típus:* Két dolgot jelent:

- milyen értékei lehetnek a változónak és
- milyen műveleteket végezhetünk vele

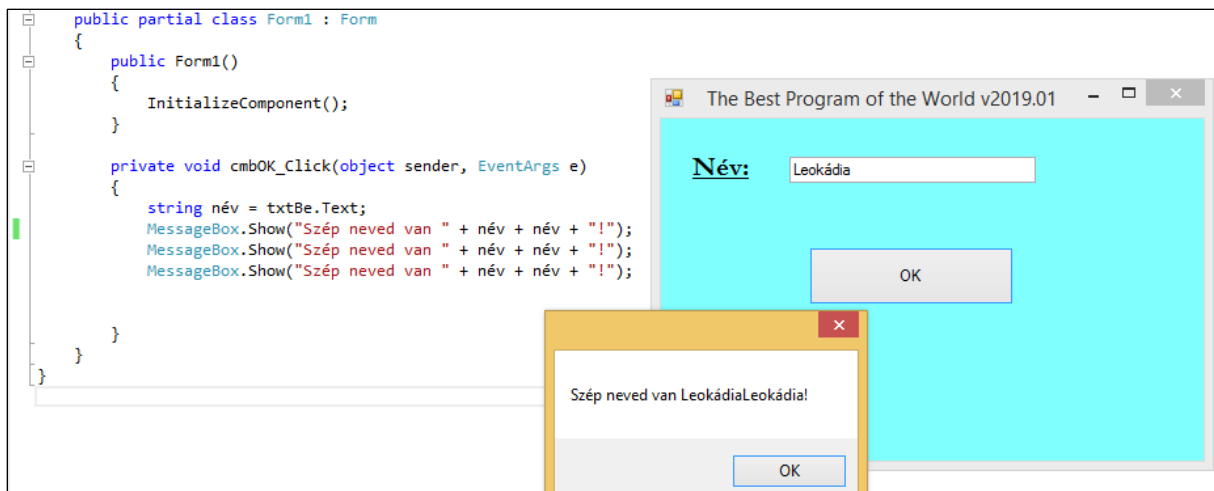
Például a `string` név; utasítás létrehoz egy név nevű, szöveges adatokat tároló változót

## Szöveges típus (Sztring)

Szöveges adatokat tárol. A legfontosabb művelet a szövegek összeadása, jele `+`, például:

„fa” + „fej” = „fafej”

„2” + „2” = „22”



```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }

    private void cmbOK_Click(object sender, EventArgs e)
    {
        string név = txtBe.Text;
        MessageBox.Show("Szép neved van " + név + név + "!");
        MessageBox.Show("Szép neved van " + név + név + "!");
        MessageBox.Show("Szép neved van " + név + név + "!");
    }
}
```

*Műveletek:*

### Konkatenáció (összefűzés)

„fa” + „fej” → „fafej”

nem kommutatív! → „fej” + „fa” → „fejfa”

(de asszociatív → „gyermek” + („könyv” + „tár”) = („gyermek” + „könyv”) + „tár”)

### Alsztring képzés

Gyermekkönyvtár.Substring(4,3)= „mek”

(balról jobbra a betűket számmal látja el, 0-val kezd, jelen esetben a 4-betű és az azt követő két betűt választja ki)

**Alma.Length=4** → (megadja hány karakterből áll a szó)

*Speciális karakterek:*

” ” → egy darab szóköz

”\n” → új sor

”\t” → tabulátor

## Egész típusok

Az egész típusok csak egész számokat tárolnak. Általában az *int* típust használjuk, ez az adatokat 32 biten tárolja (azaz kb. -2 milliárdtól kb. +2 milliárdig).

Az egész típusok esetén két szám hányadosa egész szám, az **osztás maradékát** a maradékképzéssel (jele: %) kapjuk meg. Pl.  $5/3=1$ , illetve  $5\%3=2$

*Példa (Pénzkiadó automata)*

```
namespace ATM
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            int p = Convert.ToInt32(textBox1.Text);
            int h = p / 20000;
            p = p % 20000;
            int ö = p / 5000;
            p = p % 5000;
            int ez = p / 1000;
            label15.Text = h.ToString();
            label16.Text = ö.ToString();
            label17.Text = ez.ToString();
        }
    }
}
```

*További egész típusok:*

|       |   |
|-------|---|
| byte  | 8 bit (-128 - +127)                                     |
| short | 2 bájt/16 bit (-32.000 - +32.000)                       |
| int   | 4 bájt/32 bit ( $2^{32}$ kb. -2 milliárd - +2 milliárd) |
| long  | 8 bájt/64 bit ( $2^{64}$ )                              |

## Valós típusok

A valós típusok olyan számokat tárolnak, amelyek törtrészt is tartalmaznak.

Általában a *double* típust használjuk, ez kb. 15 tizedes jegy pontossággal tárolja a számokat, a legnagyobb tárolható szám kb.  $10^{300}$ .

(Használható a kb. 6 tizedes jeggyel dolgozó, legfeljebb kb.  $10^{38}$ -ig működő *single* típus is.)

*Példa (Négyzet területe)*

```
namespace Négyzet
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void cmbOK_Click(object sender, EventArgs e)
        {
            double a = Convert.ToDouble(txtOldal.Text);
            double t = a * a;
            MessageBox.Show("A négyzet területe=" + t.ToString());
        }
    }
}
```

*Műveletek:*

A valós típusokkal kapcsolatos további műveletek és konstansok:

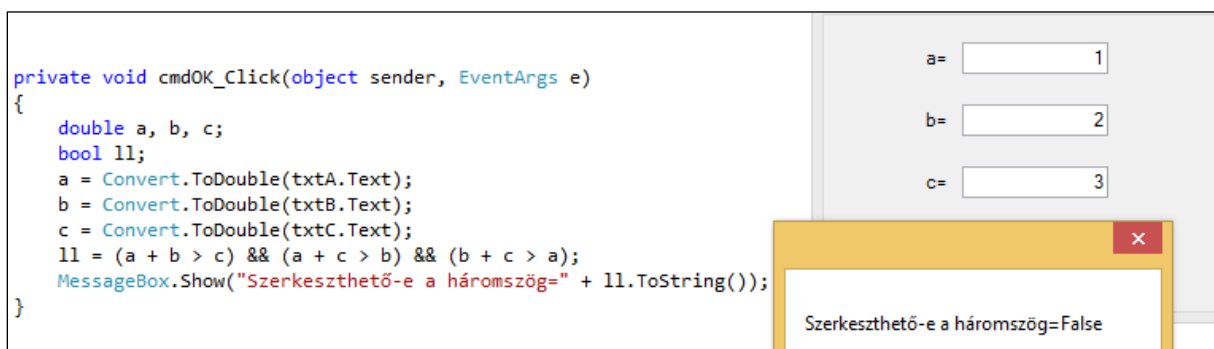
Math.PI=  $\pi$   
Math.Sqrt=  $\sqrt{x}$   
Math.Sin, Math.cos  
Math.Abs=|x|  
Math.Round = kerekít  
Math.Floor = lefelé (!) kerekít  
Math.Pow(2,3) =  $2^3$  (hatványozás)

## Logikai típus – bool

A logikai típusú változóknak két lehetséges értékük van: *true* (igaz, -1) és *false* (hamis, 0).

A logikai változókkal végezhető műveletek: *ÉS* ( && ), *VAGY* ( || ), *NEM* ( != ),  
Egyenlő-e? ( == )

*Példa (Háromszöget alkot-e három megadott pozitív szám?)*



```
private void cmdOK_Click(object sender, EventArgs e)
{
    double a, b, c;
    bool ll;
    a = Convert.ToDouble(txtA.Text);
    b = Convert.ToDouble(txtB.Text);
    c = Convert.ToDouble(txtC.Text);
    ll = (a + b > c) && (a + c > b) && (b + c > a);
    MessageBox.Show("Szerkeszthető-e a háromszög=" + ll.ToString());
}
```

The screenshot shows a Windows application window with three input fields labeled 'a=', 'b=', and 'c=' containing the values 1, 2, and 3 respectively. A yellow message box is displayed in the foreground with the text 'Szerkeszthető-e a háromszög=False'.