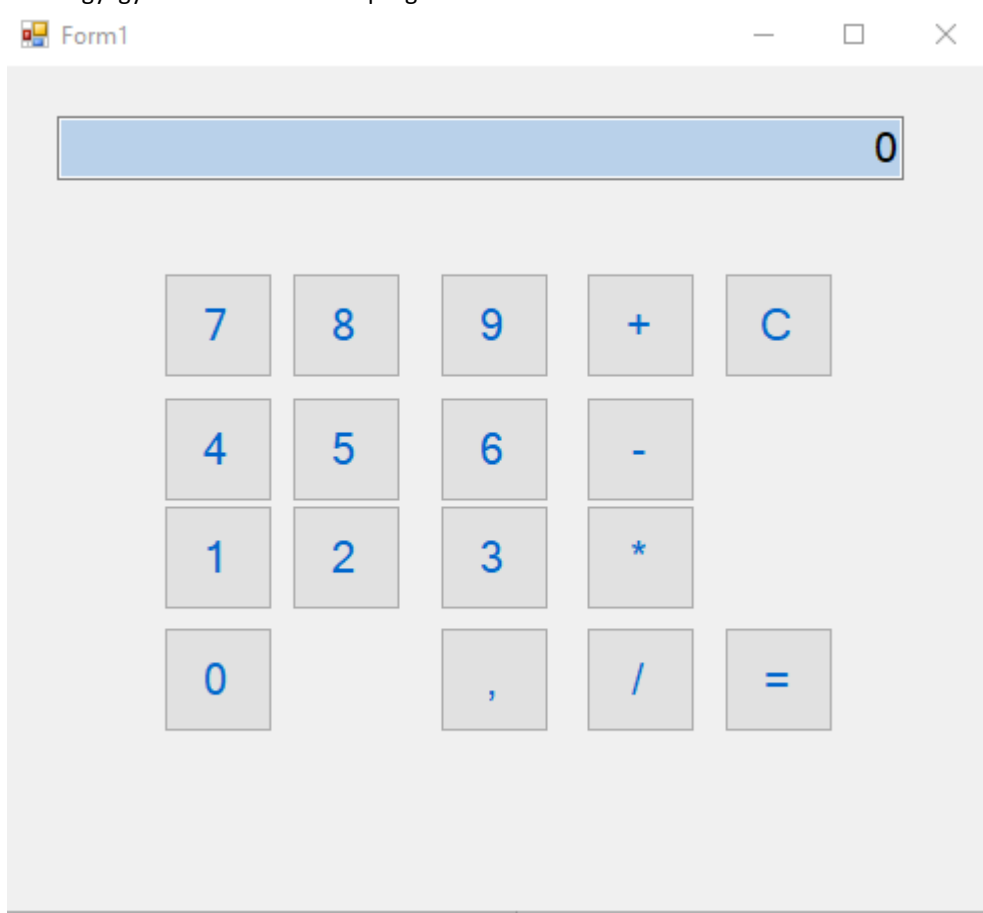


Készítsünk számológép programot!

Készítsünk egy új projektet, számológép mappába számologep néven!

Húzzunk 17 gombot és egy TextBox-ot az ablakra, 10-et a számoknak, négyet az alapműveletek, egyet a vesszőnek, egyet a C(törlés gombnak) és egyet az egyenlőségjel részére, a szövegmezőben pedig a képletetés az eredményt jelenítjük majd meg. Valahogy így kell most kinéznie a programunknak:



Jelöljük ki az összes számot tartalmazó gombot, kivéve az egyenlőségjelet tartalmazót, és rendeljük hozzájuk ugyanazt az eseménykezelőt (**Click**): **Minden számot tartalmazó gombra duplán kell kattintani!**

```
private void button1_Click(object sender, EventArgs e)
{
    textBox1.Text += ((Button)sender).Text;
}
```

Nem csinálunk itt mást mint, hogy hozzáfűzzük a szövegdobozhoz a soron következő karaktert.

**Egy hibalehetőség, hogy bizony több „operátor karaktert” és vesszőt is beírhatunk így, ezt meg kell oldani, egyszerű feltétellel.**

**Vezessünk be néhány logikai típusú változót a programunk elején!**

```
namespace számologep
{
    public partial class Form1 : Form
    {
        bool muvelet = true;
        bool vesszo = true;
        public Form1()
        {
            ...
        }
    }
}
```

Kattintsunk ezután duplán minden műveleti jelre!

```
private void button11_Click(object sender, EventArgs e)
{
    if (muvelet)
    {
        textBox1.Text += ((Button)sender).Text;
        muvelet = false;
        vesszo = true;
    }
}
```

Kattintsunk duplán a vessző gombra!

```
private void button17_Click(object sender, EventArgs e)
{
    if (vesszo)
    {
        textBox1.Text += ((Button)sender).Text;
        vesszo = false;
    }
}
```

Kattintsunk duplán a C gombra!

```
private void button16_Click(object sender, EventArgs e)
{
    textBox1.Text = "0";
    muvelet = true;
    button11.Visible = true;
    button12.Visible = true;
    button13.Visible = true;
    button14.Visible = true;
    button15.Visible = true;
}
```

Az eredményt egy külön eseménykezelőben számoljuk ki, amit az egyenlőség gombhoz rendelünk:

Az első sorban meghatározzuk az operátor indexét a stringben, hogy szétválasszthassuk a számokat, amit a második és harmadik sorban teszünk meg a **Substring** függvényel. Figyeljünk az indexekre, a második szám az operátor indexe után következik! Végül egy **switch** szerkezettel kiszámoljuk az eredményt.

```
private void button15_Click(object sender, EventArgs e)
{
    muvelet = true;
    button11.Visible = false;
    button12.Visible = false;
    button13.Visible = false;
    button14.Visible = false;
    button15.Visible = false;
    int operandus = textBox1.Text.IndexOfAny(new char[] { '+', '-', '/', '*' });
    double x = double.Parse(textBox1.Text.Substring(0, operandus));
    double y = double.Parse(textBox1.Text.Substring(operandus+1));
    double eredmeny = 0;
    switch (textBox1.Text[operandus])
    {
        case '+':
            eredmeny = x + y;
            break;
        case '-':
            eredmeny = x - y;
            break;
        case '*':
            eredmeny = x * y;
            break;
    }
}
```

```
case '/':  
    if (y != 0)  
    {  
        eredmeny = x / y;  
    }  
    else  
        textBox1.Text = "0-val nem lehet osztani";  
    break;  
}  
textBox1.Text += " = " + eredmeny;  
}
```

The screenshot shows a window titled "Form1" with a standard Windows title bar (minimize, maximize, close buttons). The main area contains a text box at the top right displaying the calculation "03\*6 = 18". Below the text box is a calculator interface consisting of a grid of buttons. The buttons are arranged as follows:

7	8	9	C
4	5	6	
1	2	3	
0		,	