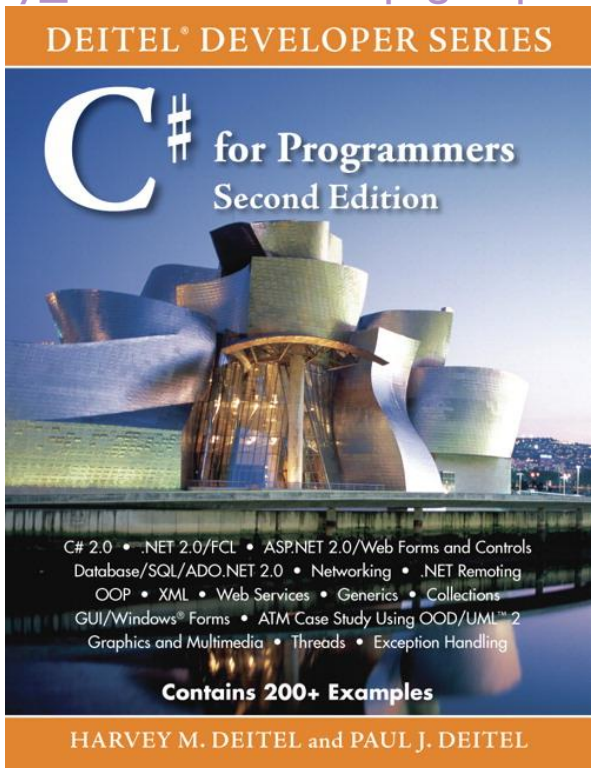
C# .Net Programming: A graphical approach
Class 1

Ing. Hazael Fernando Mojica García

- Variables
- Data Types
- Flow Control Structure **if**

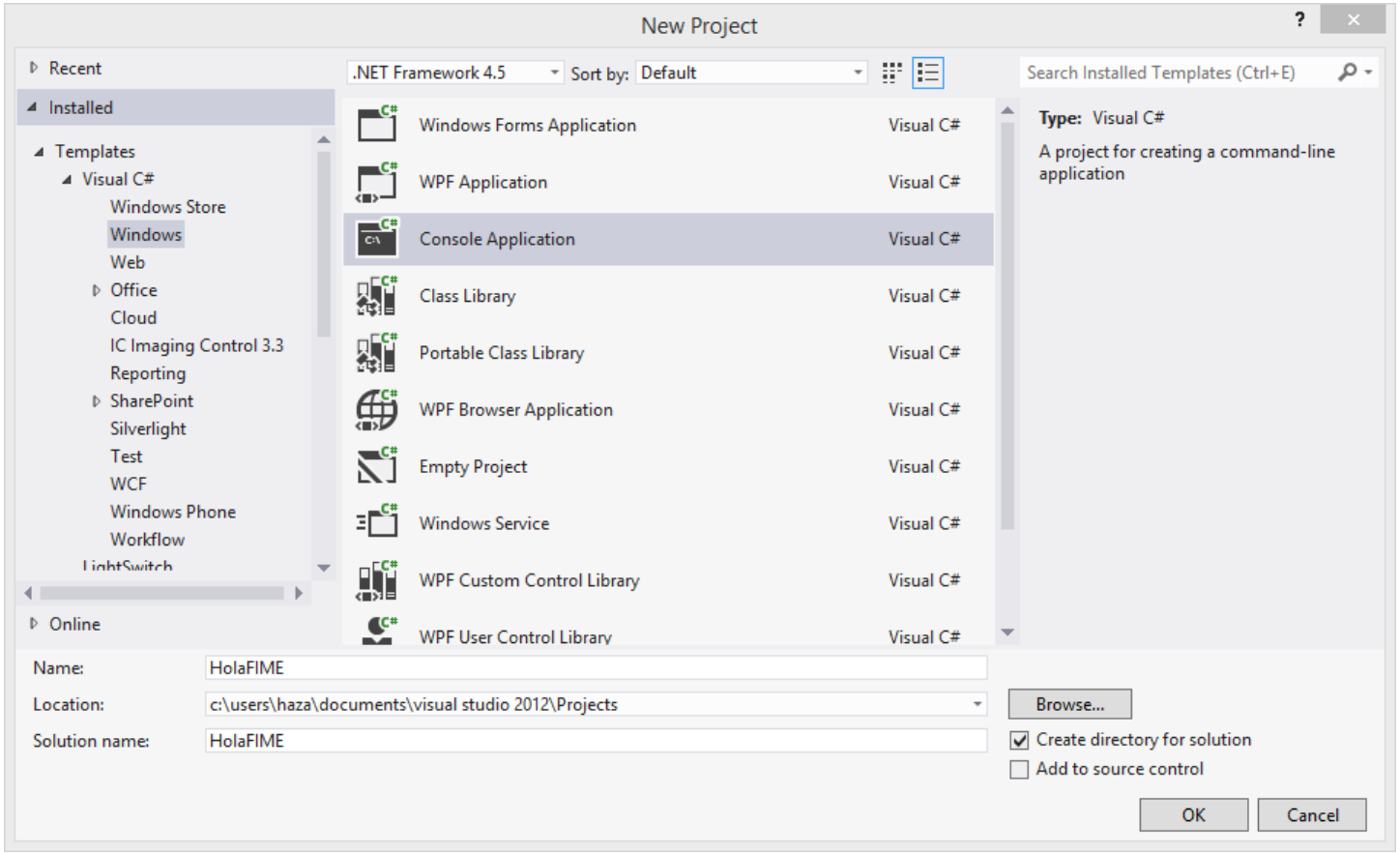
Material de Apoyo recomendado

C# Como Programar. Deitel
C# For Programmers. Deitel
https://books.google.com.mx/books?id=euV7e2f-RzsC&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false



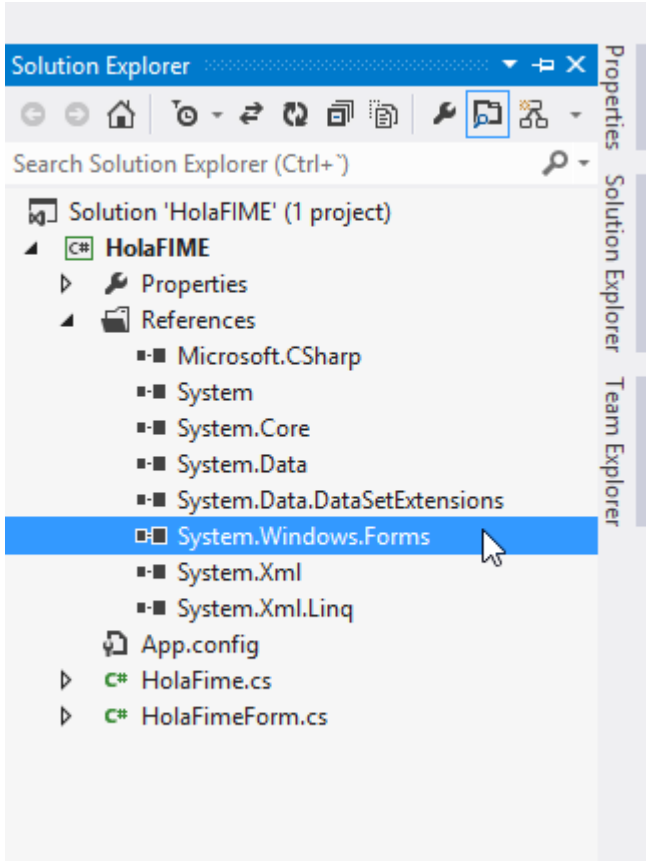
My first .Net C# pogram

Enter Visual Studio: File -> New Project



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace HolaFIME
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hola FIME");
            Console.Read();
        }
    }
}
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace HolaFIME
{
    class HolaFimeForm
    {
        static void Main(string[] args)
        {
            MessageBox.Show("Hola FIME");
        }
    }
}
```



Variables

A variable is the main object were to store values in the high level programming languages.

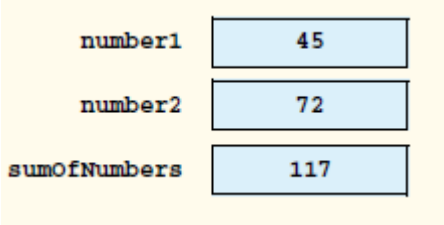
A variable is a space in memory, the smaller size is 1 byte and it could be as large as needed.

Variables allow us to store, manipulate and display data.

$$suma = number1 + number2$$

$$suma = 45 + 117$$

$$suma = 117$$



Common Data Types in C#

Short Name	.NET Class	Type	Width	Range (bits)
byte	Byte	Unsigned integer	8	0 to 255
int	Int32	Signed integer	32	-2,147,483,648 to 2,147,483,647
long	Int64	Signed integer	64	-9223372036854775808 to 9223372036854775807
float	Single	Single-precision floating point type	32	-3.402823e38 to 3.402823e38
double	Double	Double-precision floating point type	64	-1.79769313486232e308 to 1.79769313486232e308
char	Char	A single Unicode character	16	Unicode symbols used in text
bool	Boolean	Logical Boolean type	8	True or false
object	Object	Base type of all other types		
string	String	A sequence of characters		

Arithmetic Operators

Operator(s)	Operation	Order of evaluation (precedence)
( )	Parentheses	Evaluated first. If the parentheses are nested, the expression in the innermost pair is evaluated first. If there are several pairs of parentheses "on the same level" (i.e., not nested), they are evaluated left to right.
*, / or %	Multiplication Division Modulus	Evaluated second. If there are several such operators, they are evaluated left to right.
+ or -	Addition Subtraction	Evaluated last. If there are several such operators, they are evaluated left to right.

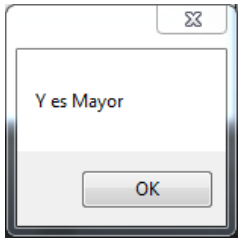
Relational Operators

Standard algebraic equality operator or relational operator	C# equality or relational operator	Example of C# condition	Meaning of C# condition
Equality operators			
=	==	x == y	x is equal to y
≠	!=	x != y	x is not equal to y
Relational operators			
>	>	x > y	x is greater than y
<	<	x < y	x is less than y
≥	>=	x >= y	x is greater than or equal to y
≤	<=	x <= y	x is less than or equal to y

Basic Logic operators

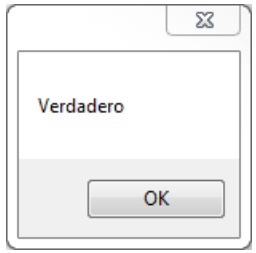
Operador Logico		Ejemplo	
&&	AND	If(x && y)	Evalua a true la expresion si, ambas variables (x, y) son true. Recordar usar && en lugar de &.
||	OR	If(x || y)	Evalua a true si cualquiera de las dos variables (x, y) es true. Recordar usar || en lugar de |

```
int x, y;
x = 30;
y = 60;

if (x > y)
{
  MessageBox.Show("X es Mayor");
}
else
{
  MessageBox.Show("Y es Mayor");
}
```

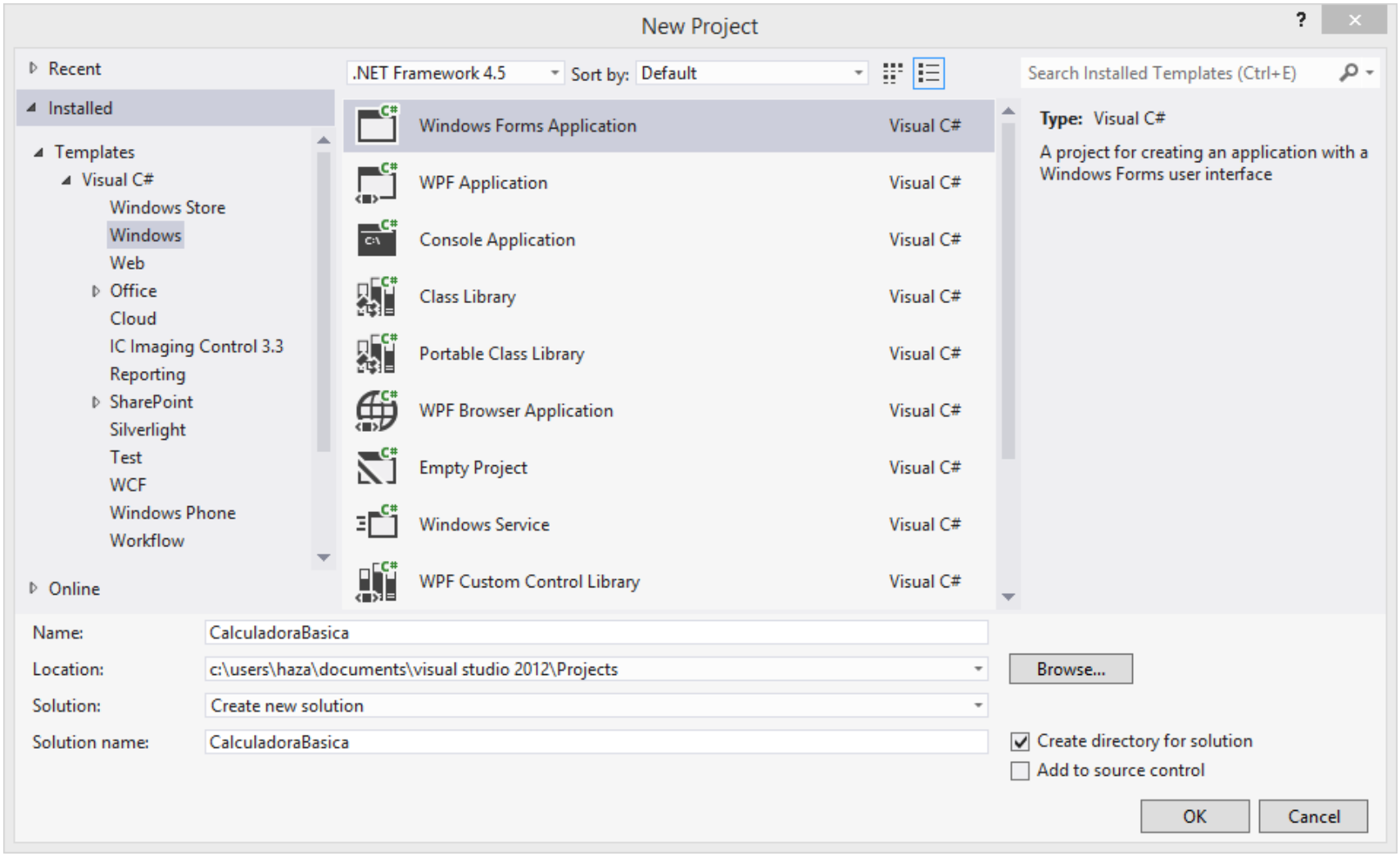



```
bool x, y;
x = true;
y = false;

if (x && y)
{
   MessageBox.Show("Verdadero");
}
else
{
   MessageBox.Show("False");
}
```


```
bool x, y;
x = true;
y = false;

if (x || y)
{
   MessageBox.Show("Verdadero");
}
else
{
   MessageBox.Show("False");
}
```

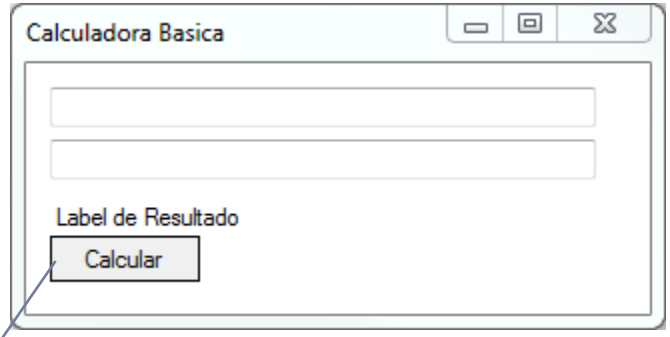


First GUI in .Net C#



Basic Calculator

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace CalculadoraBasica
{
    public partial class main : Form
    {
        public main()
        {
            InitializeComponent();
        }

        private void button_Calcular_Click(object sender, EventArgs e)
        {

        }
    }
}
```



Show the sum of the two values inside the TextBoxes

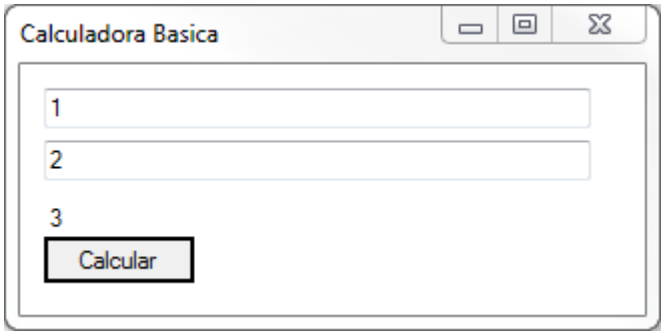
```
private void button_Calcular_Click(object sender, EventArgs e)
{
            int numero1;
            int numero2;
            int resultado;

            numero1 = Convert.ToInt32(this.textBox_Val1.Text);
            numero2 = Convert.ToInt32(this.textBox_Val2.Text);
            resultado = numero1 + numero2;
            this.label_Resultado.Text = resultado.ToString();

}
```

Variable declaration

Get each value and perform a data type convertion

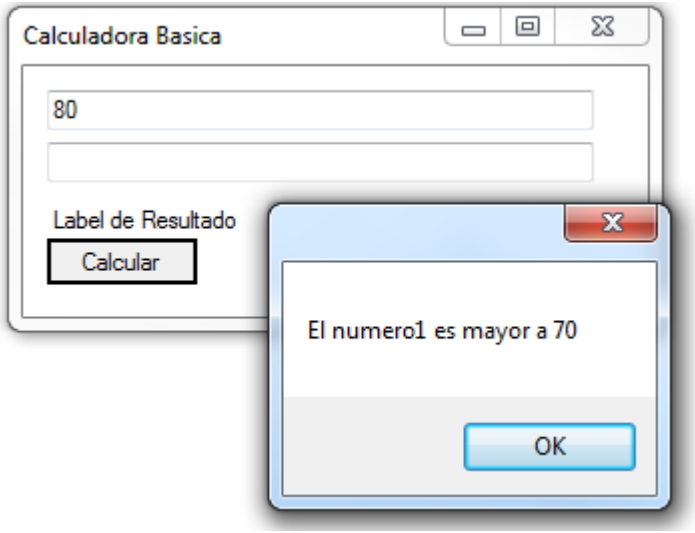
Sum

Desplegamos resultado

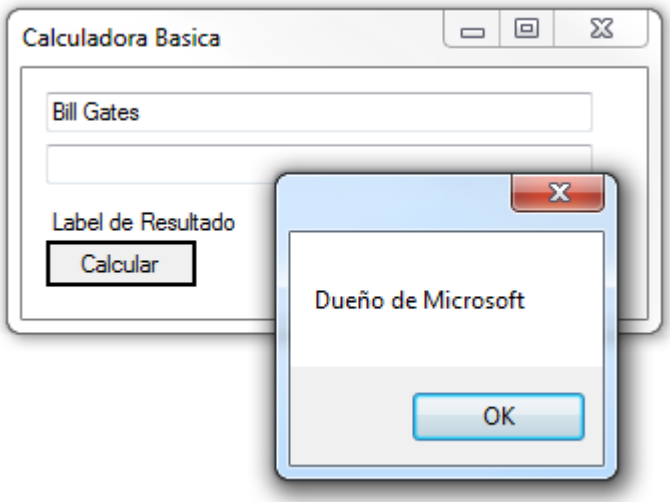


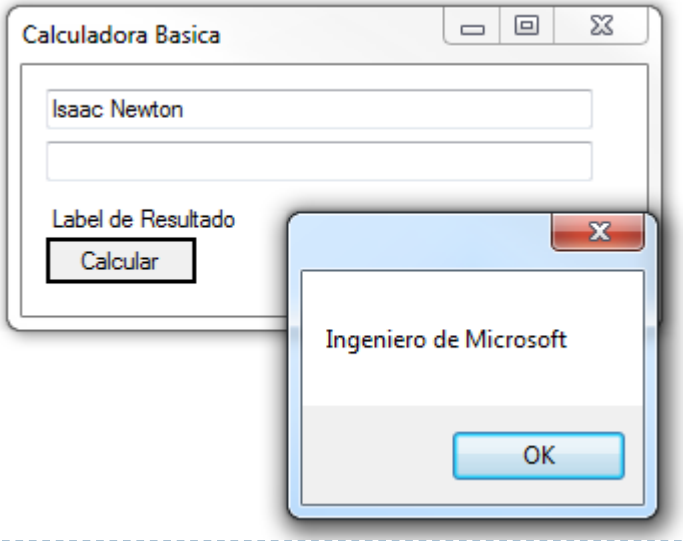
Flow Control Structure **if ... else**

```
private void button_Calcular_Click(object sender, EventArgs e)
{
    int numero1;

    numero1 = Convert.ToInt32(this.textBox_Val1.Text);

    if (numero1 > 70)
    {
        MessageBox.Show("El numero1 es mayor a 70");
    }
    else
    {
        MessageBox.Show("El numero1 es menor o igual a 70");
    }
}
```

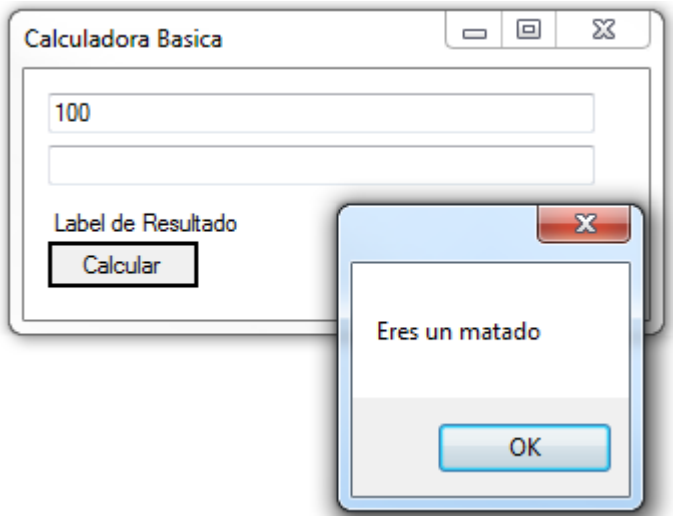


```
private void button_Calcular_Click(object sender, EventArgs e)
{
            string nombre;

            nombre = this.textBox_Val1.Text;

            if (nombre == "Bill Gates")
            {
                MessageBox.Show("Dueño de Microsoft");
            }
            else
            {
                MessageBox.Show("Ingeniero de Microsoft");
            }
}
```





```
private void button_Calcular_Click(object sender, EventArgs e)
{
            int numero1;

            numero1 = Convert.ToInt32(this.textBox_Val1.Text);

            if (numero1 >= 90)
            {
                MessageBox.Show("Eres un matado");
            }
            else
            {
                if (numero1 >= 70)
                {
                    MessageBox.Show("Panzaste");
                }
                else
                {
                    MessageBox.Show("Nos vemos en segundas");
                }
            }
}
```



1.2-CalculadoraBasica

Homework

1. Using the examples of this class change the code in order to make a program with a GUI. The GUI will have two TexBoxes and a button, you will be able to write numbers inside each TextBox, when you press the button the program will pop up a message telling which number is bigger.
2. Using the last excercise as a basis, now I want to pop up a message saying something about the sum of the two numbers:
    1. Sum is minor or equal 256 – "It fits 8 bits"
    2. Sum is between 257 and 1024 – "It fits in 10 bits"
    3. Sum is greater than 1024 – "That number is too big"
3. When the user press the button I want a message with the text of the first TextBox concatenated with the text of the second. For example: If the first textbox is "Hello " and the second "FIME" the pop up will say "Hello FIME"
4. When the user press the button I want a pop up telling me the number of characters in both TextBoxes. For example: If the first textbox is "Hello " and the second "FIME" the pop up will say **10**.