



## UNIDAD N° 2

### EJERCICIO N° 2

#### EXAMINANDO OPERADORES ARITMÉTICOS

1. Descargar el ejemplo **MathsOperators** de:  
<https://frrutneduar.sharepoint.com/:u:/r/sites/PAV2020/Documentos%20comp%20artidos/Unidad%202/Ejercicios/Codigo/MathsOperators.zip?csf=1&web=1&e=Xlwdlo>
  2. Iniciar Visual Studio 2017
  3. Ir al menú Archivo → Abrir → Proyecto / Solución
  4. Ubicar la carpeta donde se descargó el ejemplo
  5. Seleccionar la solución MathsOperators y elegir Abrir.
  6. Ir al menú Depurar → Iniciar Depuración
- Debería aparecer una ventana similar a:

MathsOperators

Operando Izquierda

Operando Derecha

☐ [+] Suma

☐ [-] Resta

☐ [\*] Multiplicación

☐ [/] División

☐ [%] Resto

Calcular

Expresión:

Resultado:

Salir

7. En el operando de la izquierda ingresar el valor 54.
8. En el operando de la derecha ingresar el valor 13.
9. Elegir cada una de las opciones y presionar calcular.  
*Expresión* toma la forma de la operación correspondiente.  
*Resultado* es siempre 0.
10. Luego de probar todas las opciones regresar al Visual Studio y detener la ejecución.

## REALIZAR LOS CÁLCULOS EN LA APLICACIÓN

1. En el Explorador de Soluciones expandir (si no lo está) el proyecto *MathsOperators* y hacer doble click en *Operadores.cs*

Aparecerá el formulario en la vista de diseño.

2. Ir al menú Ver → Otras Ventanas → Esquema del Documento  
Aparecerá la ventana Esquema del Documento. Esta ventana es útil cuando se trabaja con vistas complejas que disponen de muchos controles.
3. Al elegir un elemento del esquema se selecciona en la vista de diseño (y viceversa).
4. Para visualizar el código ir al menú Ver → Código (F7).
5. En el código ubicar el método *addValues*

```
private void addValues()
{
    int lhs = int.Parse(lhsOperand.Text);
    int rhs = int.Parse(rhsOperand.Text);
    int outcome = 0;
    //TODO: Add rhs to lhs and store the result in outcome
    expression.Text = lhsOperand.Text + " + " +
rhsOperand.Text;
    result.Text = outcome.ToString();
}
```

La primera declaración en este método declara una variable *int* llamada *lhs* y la inicializa con el entero correspondiente al valor escrito por el usuario en el cuadro *lhsOperand*. Recuerde que la propiedad *Text* de un control *TextBox* contiene una cadena, pero *lhs* es un *int*, por lo que debe convertir esta cadena en un entero antes de poder asignarla a *lhs*. El tipo de datos *int* proporciona el método *int.Parse*, que hace precisamente esto.

La segunda declaración declara una variable *int* llamada *rhs* y la inicializa al valor en el cuadro *rhsOperand* después de convertirla en un *int*.

La tercera declaración declara una variable *int* llamada *outcome*.

A continuación se incluye un comentario que indica que debe sumar *rhs* a *lhs* y almacenar el resultado en el *outcome*.

Este es el código faltante que implementará en el siguiente paso.

La quinta declaración concatena las tres cadenas que indican el cálculo que se está realizando (usando el operador más, +) y asigna el resultado a la propiedad *expression.Text*. Esto hace que la cadena aparezca en el cuadro de texto del formulario.

La declaración final muestra el resultado del cálculo al asignarlo a la propiedad *Text* del cuadro *result*. Recuerde que la propiedad *Text* es una cadena, y el resultado del cálculo es un *int*, por lo que debe convertir el *int* en una cadena antes de asignarlo a la propiedad *Text*. Recuerde que esto es lo que hace el método *ToString* del tipo *int*.

6. A continuación del comentario, en el método *addValues* agregar lo siguiente:

```
outcome = lhs + rhs;
```

7. A continuación del comentario, en el método *subtractValues* agregar lo siguiente:

```
outcome = lhs - rhs;
```

8. En los siguientes métodos agregar:

```
outcome = lhs * rhs;
```

```
outcome = lhs / rhs;
```

```
outcome = lhs % rhs;
```

9. Ir al menú Depurar → Iniciar Depuración

El formulario aparecerá nuevamente.

10. En el operando de la izquierda ingresar el valor 54.  
11. En el operando de la derecha ingresar el valor 13.  
12. Comprobar los diferentes resultados.  
13. Regresar al Visual Studio y detener la ejecución.