

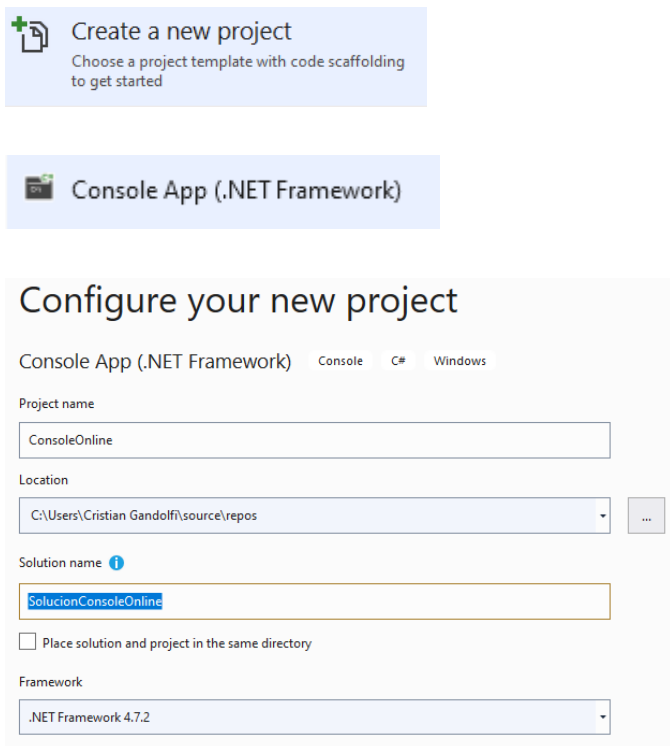
Pasos para ejecución de la actividad



1. Crear proyecto de Consola en Visual Studio que despliegue un Hola Mundo por consola

Paso 1

Abrir Visual Studio y crear un proyecto de consola con el nombre ConsoleOnline:

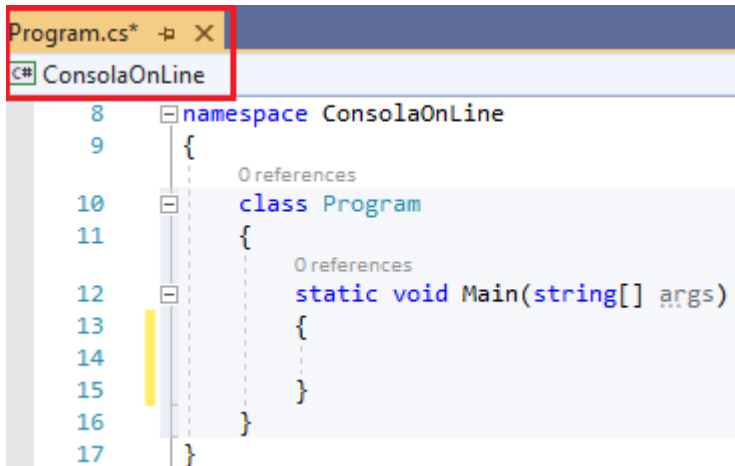


The image shows the 'Create a new project' dialog in Visual Studio. The first panel shows the 'Create a new project' button with the text 'Choose a project template with code scaffolding to get started'. The second panel shows the 'Console App (.NET Framework)' template selected. The third panel, titled 'Configure your new project', shows the following configuration:

- Project name: ConsoleOnline
- Location: C:\Users\Cristian Gandolfi\source\repos
- Solution name: SolucionConsoleOnline
- Place solution and project in the same directory
- Framework: .NET Framework 4.7.2

Nota: Es importante tener presente que una solución en Visual Studio puede contener varios proyectos y de diferentes tipos, por eso es importante diferenciar la descripción de una solución con la de un proyecto. De la misma forma un proyecto esta asociado solo a una solución.

El siguiente será el programa principal del proyecto:



```
8 namespace ConsoleOnline
9 {
10     class Program
11     {
12         static void Main(string[] args)
13         {
14         }
15     }
16 }
17 }
```

Paso 2

Crear una clase con presionando un clic derecho sobre el nombre del proyecto y presionar agregar ítem, seleccionar Class y nombrarla Saludo.

Después realizar las siguientes modificaciones destacadas en el siguiente código:

```
namespace ConsoleOnline
{
    public class Saludo
    {
        public string HolaMundo()
        {
            return "Hola Mundo Aiep";
        }
    }
}
```

Nota: Al crear una clase con visual studio podrán apreciar que generan el código base por defecto, y a este código le hemos agregado la palabra reservada public en la cabecera de la clase Saludos para que sea visible cuando creamos una instancia. También creamos un método público que devolverá un string como resultado, la palabra reservada return es la encargada de devolver el valor.

Paso 3

En el programa principal (Program.cs) ingresar el siguiente código destacado en el cuadro rojo:

```
namespace ConsolaOnLine
{
    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {
            //Creamos un objeto o instanciamos
            //un objeto de la clase Saludo
            Saludo objetoSaludo = new Saludo();
            //Creamos una variable tipo string
            string mensaje;
            //Le asignamos a la variable mensaje
            //el valor que retorna el metodo
            //HolaMundo() de la clase Saludo
            mensaje = objetoSaludo.HolaMundo();
            //Ahora imprimiremos el valor de la variable mensaje
            //utilizamos la instrucción Console.WriteLine
            //y Console.Read() para generar una pausa
            Console.WriteLine(mensaje);
            Console.Read();
        }
    }
}
```

Nota: En las líneas de código donde se anteponen dos slash (//) se ha comentado el código, en los comentarios esta la explicación línea por línea de lo que se está realizando.

Paso 4

Presionamos la opción para ejecutar y se visualizará el siguiente resultado:

```
Hola Mundo Aiep
```

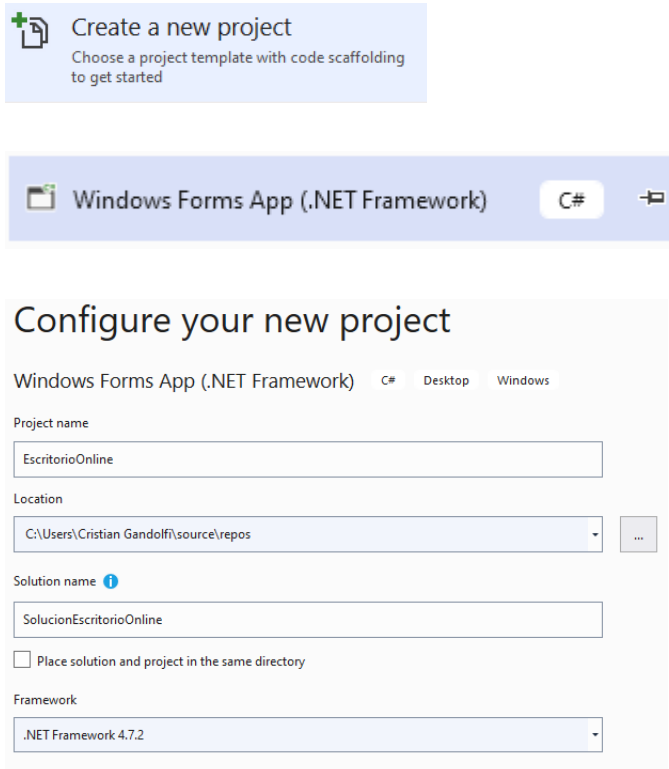
Paso 5

Los alumnos deben optimizar el código, buscando la alternativa que consideren mejor para obtener el mismo resultado:

2. Crear proyecto de Escritorio en Visual Studio que evalúe si un número es par o impar

Paso 1

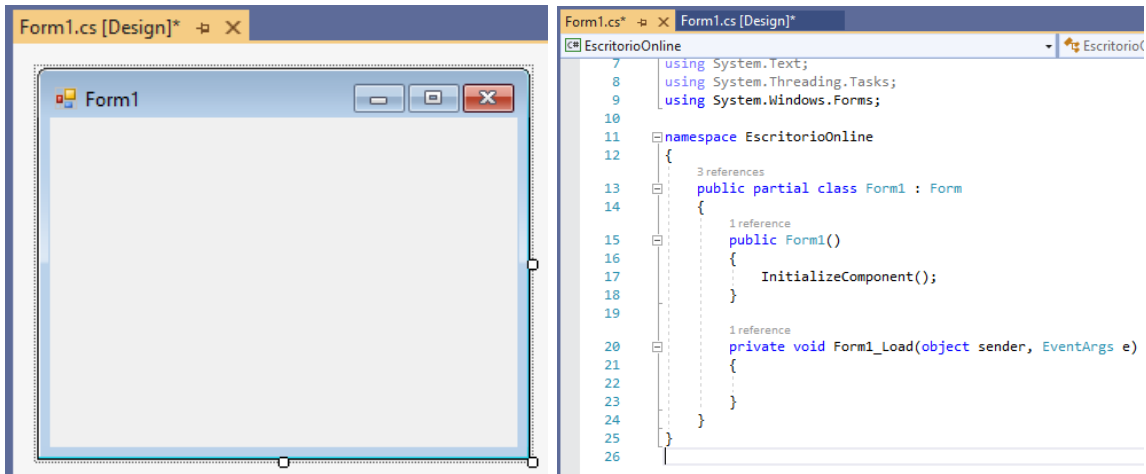
Abrir Visual Studio y crear un proyecto de escritorio con el nombre EscritorioOnline:



The image shows the 'Create a new project' dialog in Visual Studio. The first part is a blue banner with a plus icon and the text 'Create a new project' and 'Choose a project template with code scaffolding to get started'. Below this is a selection bar for 'Windows Forms App (.NET Framework)' with a C# icon and a plus icon. The second part is the 'Configure your new project' dialog. It has tabs for 'C#', 'Desktop', and 'Windows'. The 'Project name' field contains 'EscritorioOnline'. The 'Location' field contains 'C:\Users\Cristian Gandolfi\source\repos'. The 'Solution name' field contains 'SolucionEscritorioOnline'. There is an unchecked checkbox for 'Place solution and project in the same directory'. The 'Framework' dropdown is set to '.NET Framework 4.7.2'.

Nota: Es importante tener presente que una solución en Visual Studio puede contener varios proyectos y de diferentes tipos, por eso es importante diferenciar la descripción de una solución con la de un proyecto. De la misma forma un proyecto está asociado solo a una solución.

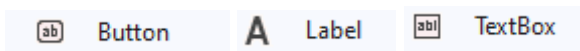
El siguiente será el programa principal del proyecto:



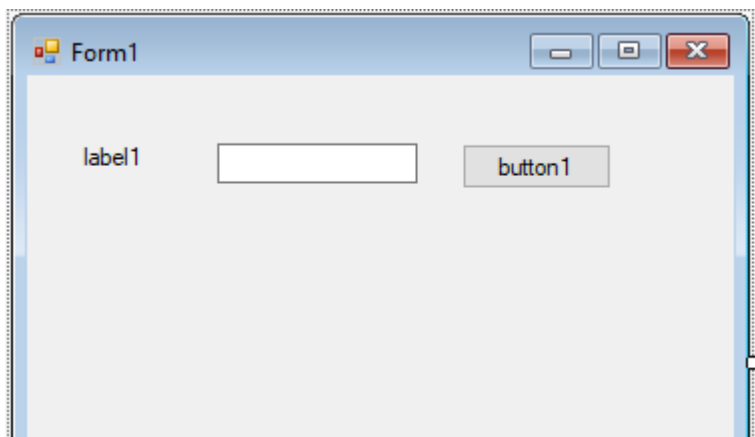
En la imagen de la izquierda podemos visualizar el formulario en modo diseño y al lado derecho el programa principal o la clase asociada al formulario.

Paso 2

En el formulario ingresaremos los siguientes elementos disponibles desde la barra de herramientas: TextBox, Label y Button.



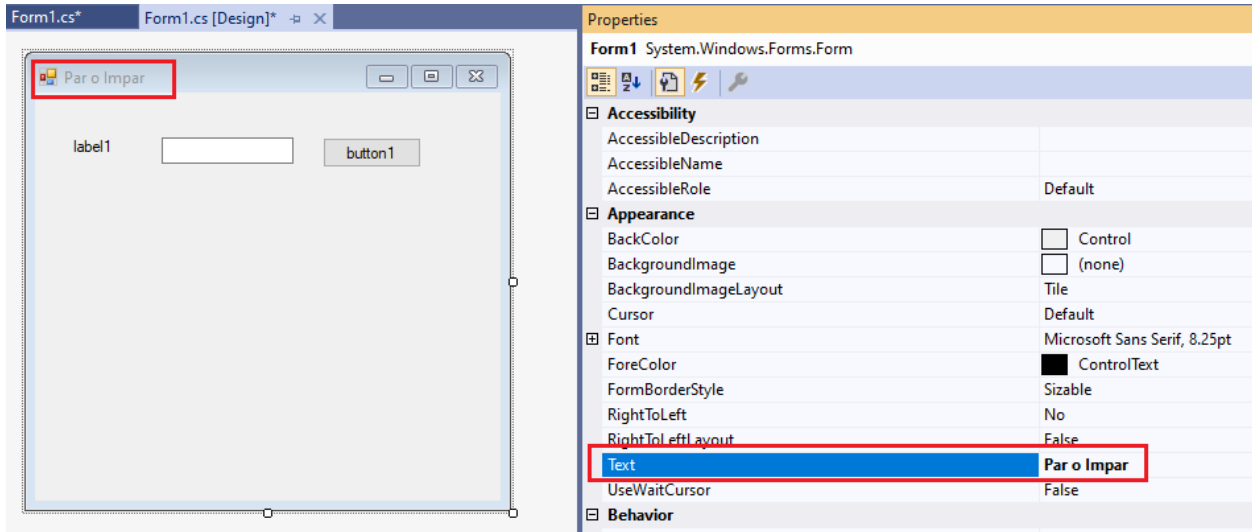
El formulario debe quedar de la siguiente forma:



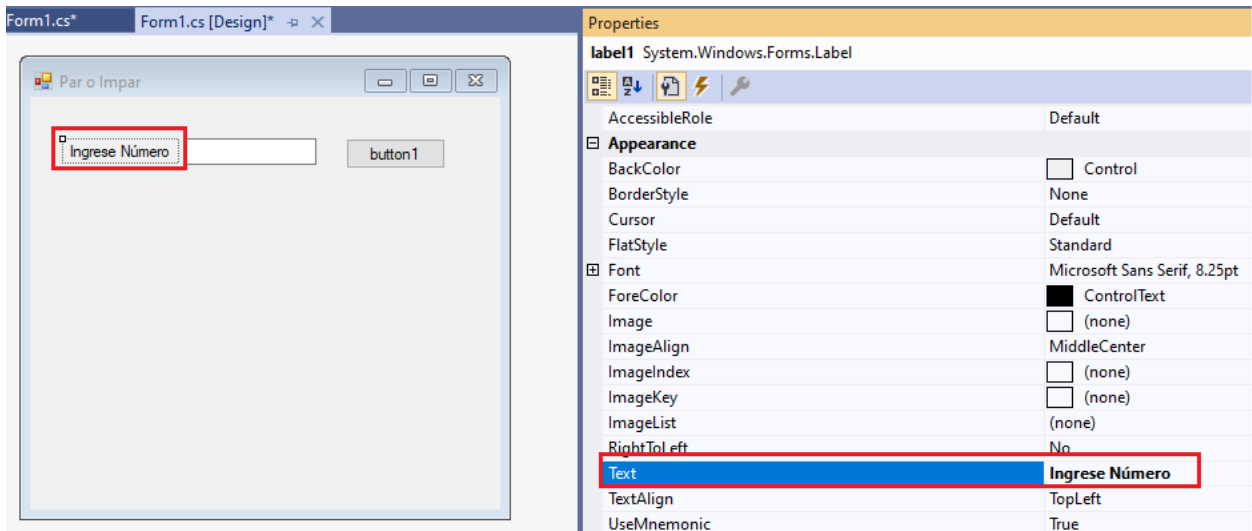
Paso 3

Para modificar las propiedades de los elementos debemos seleccionar cada elemento y modificar algunas propiedades.

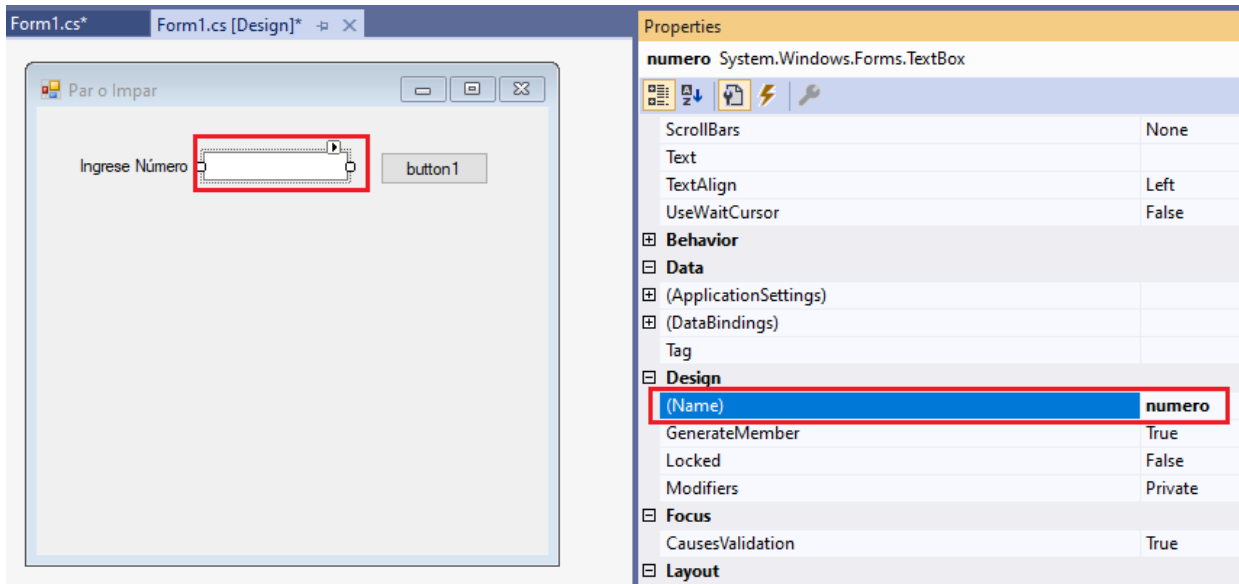
Seleccionamos el formulario y modificaremos la propiedad Text como se visualiza en la imagen:



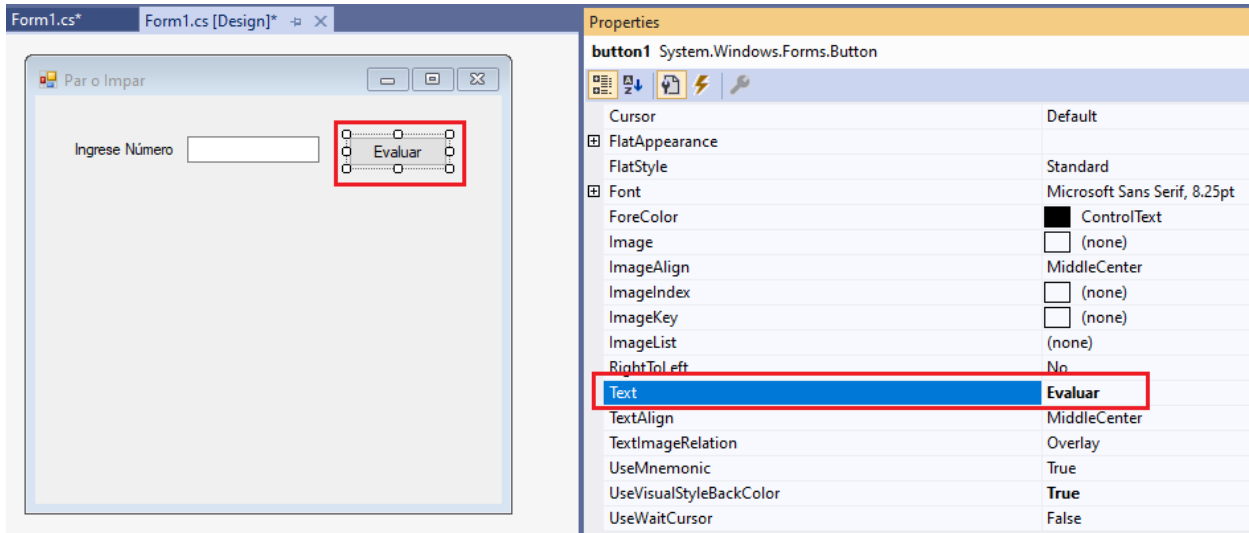
Seleccionamos el Label y modificaremos la propiedad Text como se visualiza en la imagen:



Seleccionamos el TextBox y modificaremos la propiedad Name como se visualiza en la imagen:



Seleccionamos el Button y modificaremos la propiedad Text como se visualiza en la imagen:



Paso 4

Crear una clase con presionando un clic derecho sobre el nombre del proyecto y presionar agregar ítem, seleccionar Class y nombrarla Operaciones.

Después realizar las siguientes modificaciones destacadas en el siguiente código:

```
public class Operaciones
{
    //metodo que retornara true o false
    0 references
    public bool parImpar(int num)
    {
        //En la siguiente línea de código preguntamos si el
        //resto (%) de la división del número recibido como
        //parametro es 0 o no
        if(num % 2 == 0)
        {
            return true; //si es 0 signifca que el número es par
            //y el metodo retornará true
        }
        else
        {
            return false;//si es distinto a 0 signifca que el
            //número es impar y el metodo retornará false
        }
    }
}
```

Nota: En las líneas de código donde se anteponen dos slash (//) se ha comentado el código, en los comentarios esta la explicación línea por línea de lo que se está realizando.

Paso 5

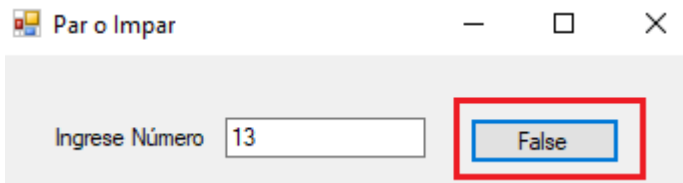
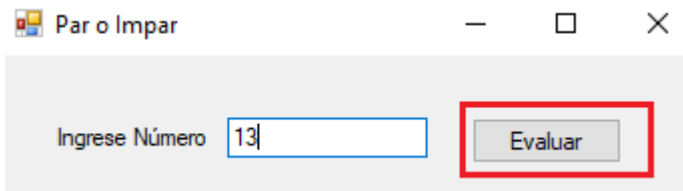
Ahora iremos al programa principal o la clase del formulario e ingresaremos el siguiente código:

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

    1 reference
    private void Form1_Load(object sender, EventArgs e)
    {
        //Crearemos una instancia de la clase Operaciones (objeto)
        Operaciones ObjetoEvaluacion = new Operaciones();
        //Asignamos el valor que retornara el metodo al button1 (boton) en su propiedad text
        button1.Text = Convert.ToString(ObjetoEvaluacion.parImpar(Convert.ToInt32(numero.Text)));
        //Notar que el metodo recibe un parametro entero y por eso debemos hacer la conversión con
        //Convert.ToInt32() y también debemos realizar la conversión del valor boolean que retornara
        //el metodo con Convert.ToString()
    }
}
```

Paso 6

Presionamos la opción para ejecutar, ingresamos el número 14 y se visualizará el siguiente resultado:



Nota: La respuesta es asignada al botón mediante la propiedad Text.

Paso 7

Los alumnos deben optimizar el código, creando una alternativa para ver el resultado en un elemento Label y que indique lo siguiente: Si es par "El número es par", si es impar "El número es impar".