

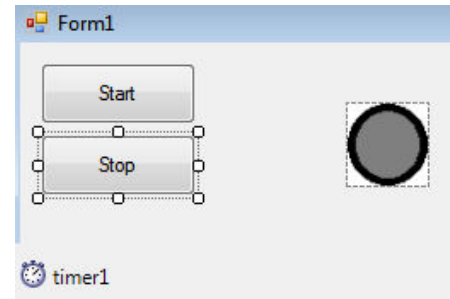
Ejercicio C# Windows Forms: Pong

En este ejercicio utilizaremos las propiedades *Left* y *Top* y el objeto *Timer* para desplazar una imagen por la ventana.

- Crea un nuevo proyecto:
Archivo – Nuevo proyecto (File – New Project)
Aplicación de Windows Forms: **Pong**
- Añade al formulario los objetos de la figura. →

Ingredientes:

- 2 Buttons: Text: Start y Stop
- 1 Timer: Interval: 20.
Image: Importar pelota.gif
- 1 Picture box. Name: bola



- Añade el código:

Primero crearemos dos variables públicas antes de public Form():

```
int vel=5;           //variable numérica entera para la velocidad
int direc = 1;      // variable numérica entera para la dirección
public Form1()
```

- Añadir el código al evento Click del botón Start: `timer1.Enabled = true;`
- Añadir el código al evento Click del botón Stop: `timer1.Enabled = false;`
- Añadir el código al evento Tic del Timer:

```
private void timer1_Tick(object sender, EventArgs e)
{
    Try //bloque a evaluar en caso de error...
    {
        if (direc == 1) //abajo der
        {
            bola.Left = bola.Left + vel;
            bola.Top = bola.Top + vel;
            if ((bola.Top+bola.Height) >= this.Height) direc = 2;
            if ((bola.Left+ bola.Width) >= this.Width) direc = 3;
        }
        if (direc == 2) //arriba der
        {
            bola.Left = bola.Left + vel;
            bola.Top = bola.Top - vel;
            if ((bola.Top) <=0) direc = 1;
            if ((bola.Left + bola.Width) >= this.Width) direc = 4;
        }
        if (direc == 3) //abajo iz
        {
            bola.Left = bola.Left - vel;
            bola.Top = bola.Top + vel;
            if ((bola.Top) >= this.Height) direc = 4;
            if ((bola.Left)<=0 ) direc = 1;
        }
        if (direc == 4) //arriba iz
        {
            bola.Left = bola.Left - vel;
            bola.Top = bola.Top - vel;
            if ((bola.Top) <=0) direc = 3;
            if (bola.Left <= 0) direc = 2;
        }
    }
    Catch // bloque en caso de error
    {
        MessageBox.Show("error de dirección");
        Close();
    }
}
```

Ejercicio propuesto:

Al hacer clic sobre la pelota, aumentará la velocidad y la puntuación.

- Incrementando la variable vel: `vel++;`
- Reduciendo el intervalo del timer: `timer1.interval--;`

Teoría: Control de errores o excepciones **TRY - CATCH - FINALLY**

Agrupar instrucciones en un bloque **try** nos permite saltar el bloque en caso de error y detectar el tipo de excepción: `outOfMemory`, `stackOverflow`, `indexOutOfRangeException`, `divideByZero`, etc...

```
try
{ instrucciones }
catch
{ instrucciones de excepción }
finally
{ instrucciones de liberación }
```