



Controles que vamos a ver hoy

PictureBox

El PictureBox es un objeto que nos sirve para contener imágenes, ya sea que nosotros directamente dibujemos sobre él o bien que la imagen la importemos de otra parte.

Principales propiedades

Name. Permite asignarle un nombre al control. Se recomienda anteponer un **pbx**Nombre.

Image. Carga o importa una imagen desde otra carpeta.

SizeMode. Permite ajustar el tamaño del objeto al de la imagen importada.

Principales métodos

Click. Ocurre cuando se da clic con el mouse.

MouseMove. Ocurre cuando el mouse se mueve sobre el objeto.

MouseDown. Ocurre cuando se oprime el botón del mouse sobre el objeto.

MouseUp. Ocurre cuando se libera el botón del mouse sobre el objeto.

Timer

El timer nos permite ejecutar un proceso cada determinado número de milisegundos.

Principales propiedades

Name. Permite asignarle un nombre al control. Se recomienda anteponer un **tmr**Nombre.

Interval. Indica el tiempo en milisegundos que deberá esperar para volver a ejecutar un proceso.

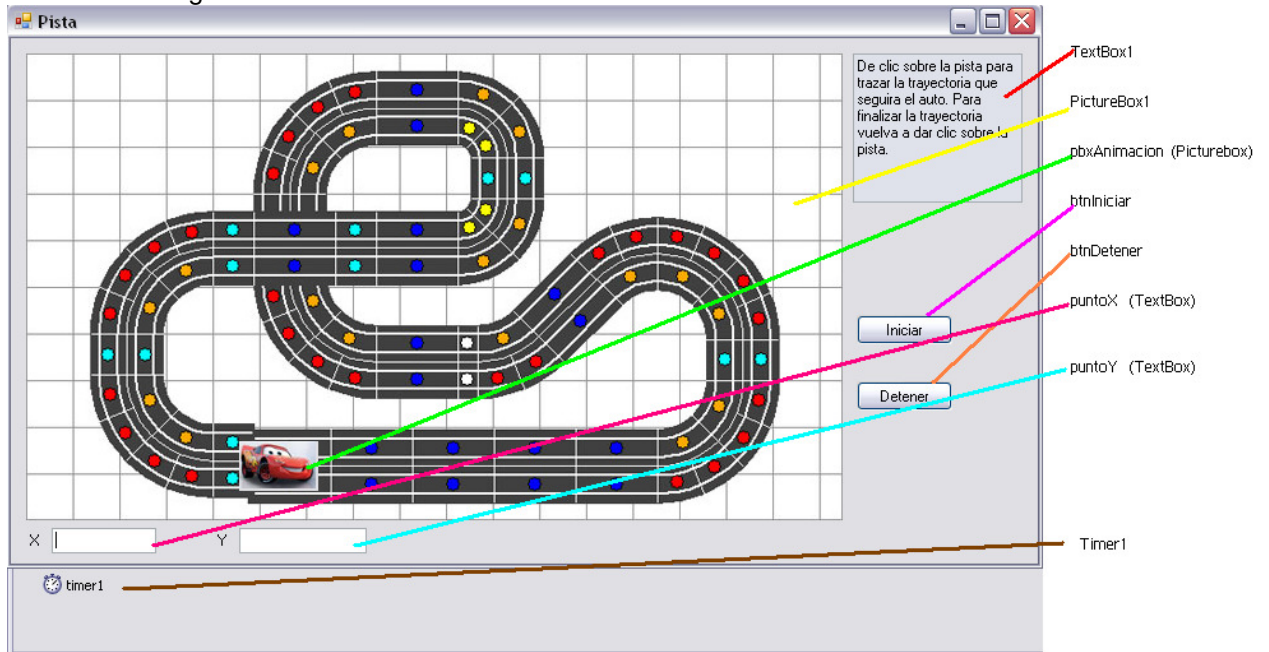
Principales métodos

Tick. Es el proceso que ocurre periódicamente, este es definido por el interval.

Ejercicios de la práctica

En esta práctica vamos a realizar una animación por medio de pictureBox y un Timer.

1. Crea un nuevo proyecto y llama a la solución animaciones y al proyecto carreras.
2. Diseña el siguiente formulario.



3. Importa imágenes para cada uno de los controles.
4. Crea una clase coordenadas.cs y adiciona el siguiente código:

```
using System;
using System.Collections.Generic;
using System.Text;
```

```
namespace carreras
{
    class coordenadas
    {
        private int x;
        private int y;

        public int X
        {
            get
            {
                return x;
            }
            set
            {
                x = value;
            }
        }

        public int Y
        {
```



TEMA4.PROGRAMACIÓN ORIENTADA A OBJETOS

```
    get
    {
        return y;
    }
    set
    {
        y = value;
    }
}
public coordenadas(int _x, int _y)
{
    this.X = _x;
    this.Y = _y;
}
}
```

5. Edita el formulario Form1.cs

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

namespace carreras
{
    public partial class Form1 : Form
    {
        bool trazar_coordenadas;
        List<coordenadas> puntos = new List<coordenadas>(); //Se crea una Lista llamada puntos
        int contador;

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            trazar_coordenadas = false;
        }

        private void pictureBox1_MouseMove(object sender, MouseEventArgs e)
        {
            if (trazar_coordenadas == true)
                puntos.Add(new coordenadas(e.X, e.Y));
            puntoX.Text = e.X.ToString();
            puntoY.Text = e.Y.ToString();
        }

        private void pictureBox1_Click(object sender, EventArgs e)
        {
            if (trazar_coordenadas == false)
            {
                puntos.Clear();
                trazar_coordenadas = true;
                btnIniciar.Text = "Grabando";
                btnIniciar.Enabled = false;
                btnDetener.Enabled = false;
            }
            else
            {

```



TEMA4.PROGRAMACIÓN ORIENTADA A OBJETOS

```
trazar_coordenadas = false;
btnIniciar.Text = "Iniciar";
btnIniciar.Enabled = true;
}
}

private void btnIniciar_Click(object sender, EventArgs e)
{
    contador = 0;
    btnDetener.Enabled = true;
    btnIniciar.Enabled = false;
    timer1.Start();
}

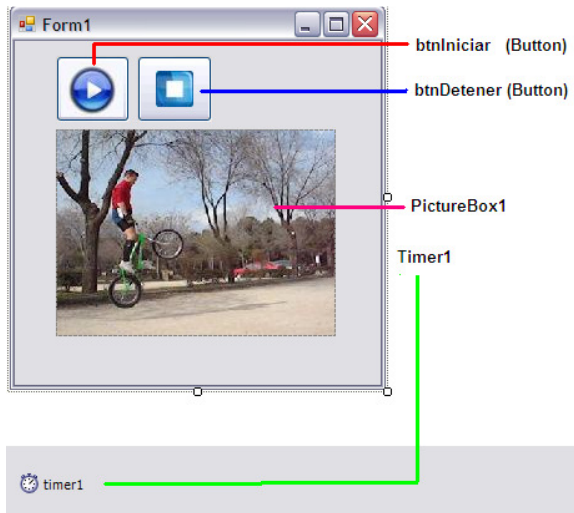
private void timer1_Tick(object sender, EventArgs e)
{
    contador++;
    if (contador < puntos.Count)
    {
        pbxAnimacion.Location = new System.Drawing.Point(puntos[contador].X, puntos[contador].Y);
    }
    else
        timer1.Stop();
}

private void btnDetener_Click(object sender, EventArgs e)
{
    btnIniciar.Enabled = true;
    btnDetener.Enabled = false;
    timer1.Stop();
}
}
```

6. Compíllalo, ejecútalo y anota tus comentarios. **NO OLVIDES DECLARAR LOS EVENTOS!!!**

TEMA4.PROGRAMACIÓN ORIENTADA A OBJETOS

7. Crea un nuevo proyecto y llámalo secuencia_imagenes
8. Diseña el formulario de la siguiente manera:



9. Edita el formulario Form1.cs

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

```
namespace secuencia_imagenes
```

```
{
    public partial class Form1 : Form
```

```
    {
        int contador;
        public Form1()
        {
            InitializeComponent();
        }
    }
```

```
    private void btnIniciar_Click(object sender, EventArgs e)
    {
        timer1.Start();
    }
```

```
    private void btnDetener_Click(object sender, EventArgs e)
    {
        timer1.Stop();
    }
}
```



TEMA4.PROGRAMACIÓN ORIENTADA A OBJETOS

```
private void timer1_Tick(object sender, EventArgs e)
{
    contador++;
    if (contador <= 16)
        pictureBox1.Load("../imagenes/salto" + contador + ".jpg");
    else
        contador = 1;
}

private void Form1_Load(object sender, EventArgs e)
{
    contador = 1;
}
}
```

10. NO OLVIDES crear los eventos en el Form1.Designer.cs
11. Deberás adicionar al proyecto la carpeta imágenes
12. Compílalo, ejecútalo y anota tus comentarios