

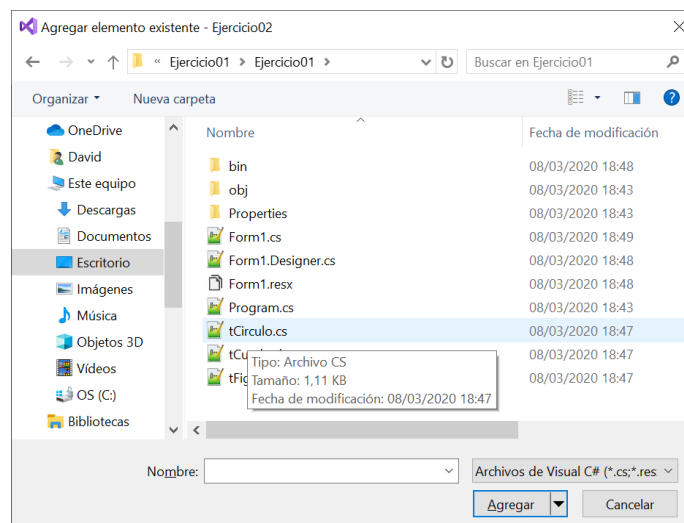
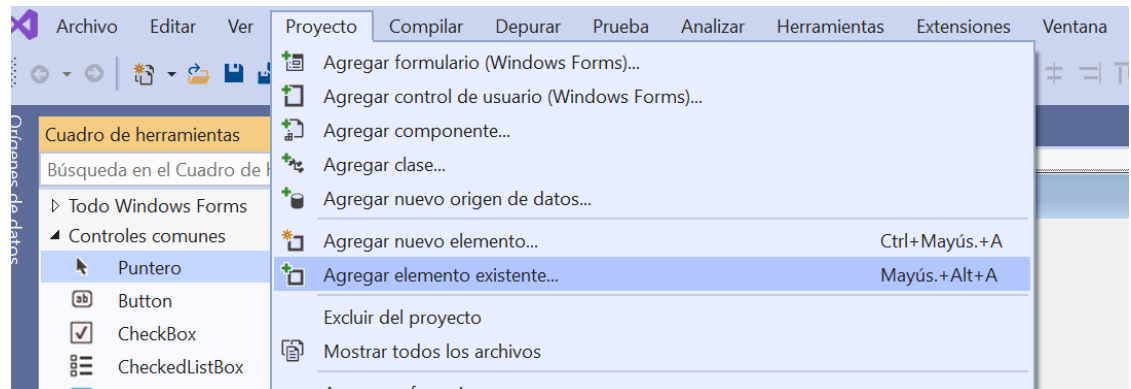
1. Realizar un proyecto en Visual Studio en el que tengáis la clase base **Figura**, y las clases **Circulo** y **Cuadrado** que heredan de Figura.

Podéis seguir los pasos que indica el tema hasta la mitad de la página 12. Tendréis un formulario con dos botones en los que en cada uno de ellos se cree un objeto de tipo Circulo o Cuadrado y se muestren sus datos.

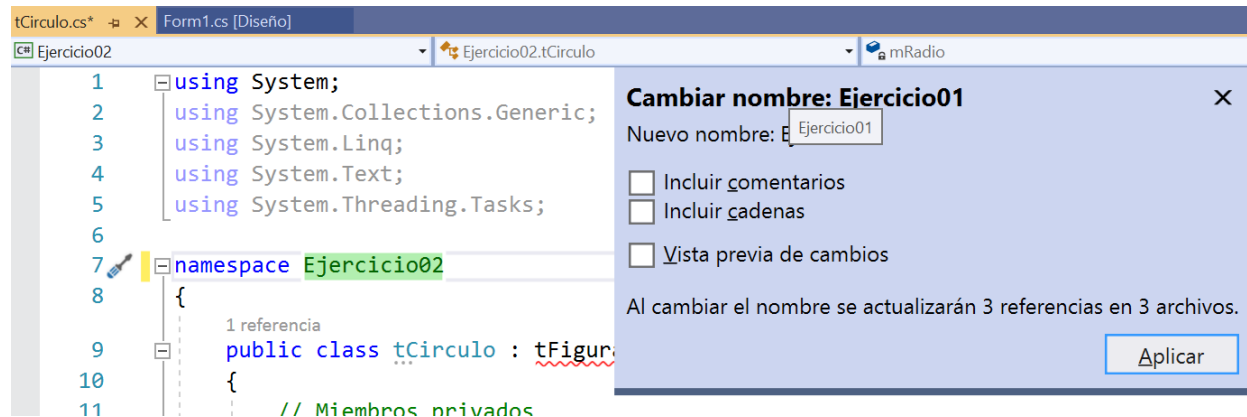
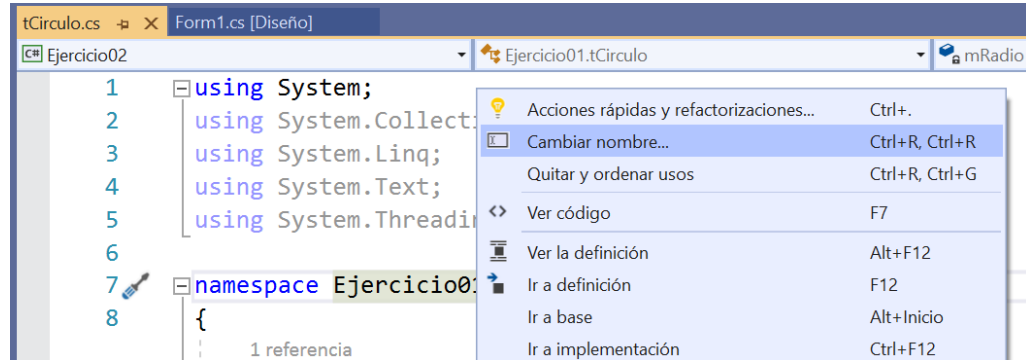
2. Realizar un **nuevo ejercicio** en el que utilicéis las clases Figura, Circulo y Cuadrado del ejercicio anterior.

Si queremos utilizar las clases del ejercicio anterior, podemos hacer varias cosas:

- Podemos, en el explorador de Windows copiar la carpeta del proyecto del ejercicio 1 y modificar el nombre de la carpeta a ejercicio 2. De esta forma tenemos las clases y empezamos a modificar para hacer este ejercicio 2.
- Otra posibilidad es crear un nuevo proyecto para el ejercicio 2 e incorporar las clases de ejercicio 1. Esto se hace en el menú **Proyecto -> Agregar elemento** existente y eligiendo las clases Figura, Cuadrado y Circulo del ejercicio 2.



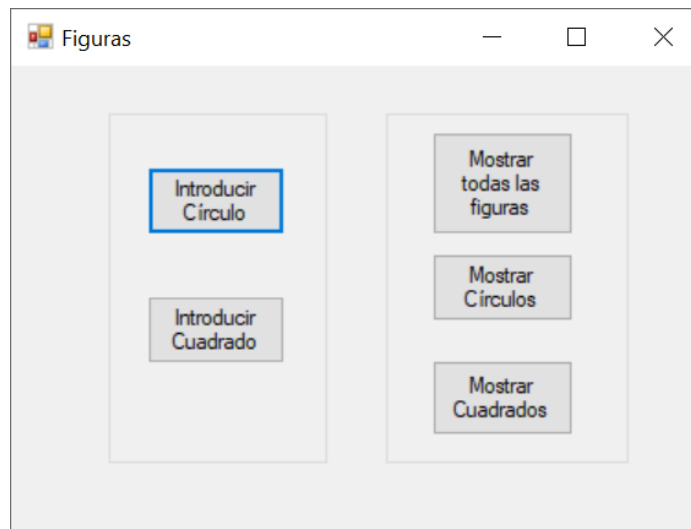
Con esto hemos conseguido incorporarlas a nuestro proyecto. Para poder utilizarlas debemos cambiar además el **namespace** (espacio de trabajo) de las clases. Para ello entramos en alguna de esas clases y donde pone namespace Ejercicio01 pulsamos botón derecho Cambiar nombre... y lo cambiamos por Ejercicio02.



A partir de este momento ya podemos utilizar las clases en nuestro proyecto.

Continúa en la página siguiente...

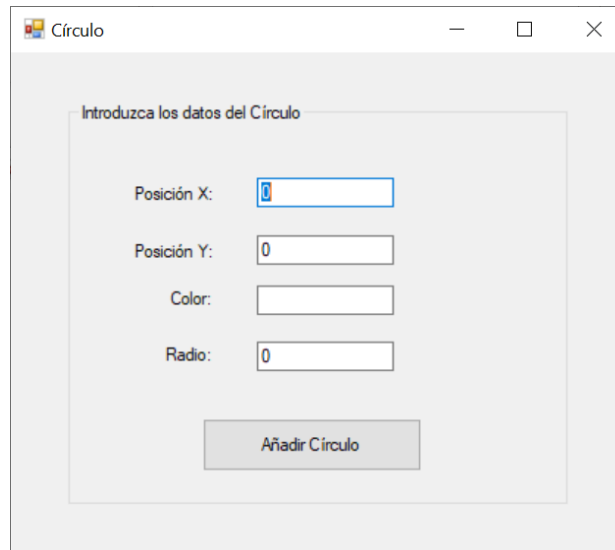
Ahora seguiremos las indicaciones de tema desde la página 12 hasta la 17 para realizar un formulario que nos permita ir introduciendo cuadrados y círculos en una lista, con botones para mostrar los datos de todas las figuras, mostrar únicamente los círculos y únicamente los cuadrados.



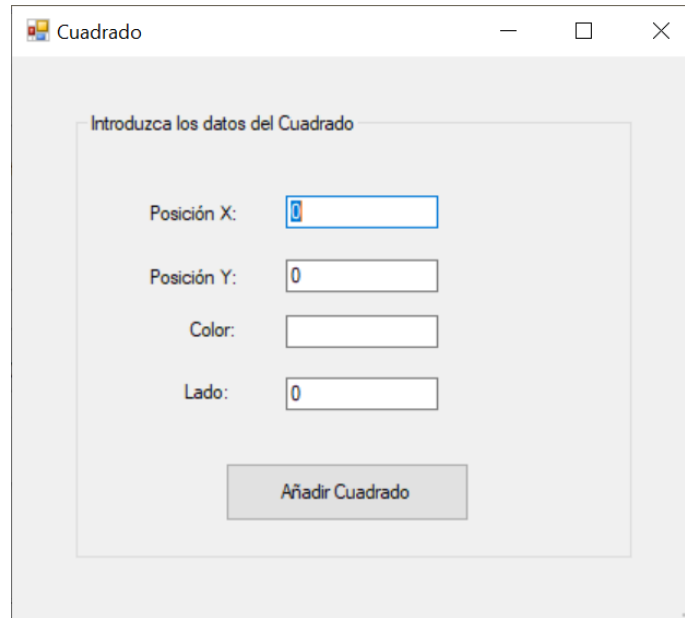
Este ejercicio lo podéis hacer siguiendo el tema hasta la última página.

3. Ampliar el ejercicio anterior (también lo podéis copiar para dejar el ejercicio 2 sin tocar) de manera que el formulario inicial sea el mismo, pero hagamos formularios específicos para introducir los datos de círculos y de cuadrados.

Cuando pulsemos el botón de **Introducir Círculo** aparecerá un formulario donde introduciremos los datos del círculo (en vez de hacerlo con InputBox como en el ejer 2):



Cuando pulsemos el botón de Introducir Cuadrado aparecerá un formulario para introducir los datos del cuadrado:



Tened en cuenta que a estos formularios deberemos pasarle el **List** de figuras que creamos en el formulario inicial, de manera parecida a como lo hacíamos en el ejercicio 6 del tema anterior...

4. Ampliar el ejercicio anterior para que entre los datos que mostramos de cada figura tengamos también el **Perímetro de la figura**.

Este ejercicio además de cuadrados y círculos nos permitirá introducir **triángulos equiláteros, rectángulos y hexágonos regulares**:

Aquí tenéis enlaces para calcular las áreas de un [triángulo](#), [rectángulo](#) y [hexágono regular](#).

5. Realizar un programa **similar al ejercicio 6 del tema anterior**. El aspecto y funcionamiento del programa será el mismo. Pero ahora tendremos las siguientes clases:
 - **Curso** donde guardaremos la información de un curso. Los datos de un curso serán el nombre y el código.
 - **ListaCursos** será una clase donde guardemos la lista de cursos de la misma manera que lo hacíamos en la actividad del tema anterior.
 - **Persona** será una **clase abstracta** de la cual heredarán otras. Los datos que tendrá la clase Persona serán el nombre, el dni y el teléfono.

- **Profesor** será una clase que heredará de Persona. La clase Profesor añadirá los siguientes datos: lista de asignaturas que imparte, código del curso del que es tutor y correo electrónico.
- **Alumno** también heredará de Persona. La clase Alumno añadirá los siguientes datos: código del curso al que pertenece, lista de notas (lista de valores double).
- **ListaPersonas** será una clase donde guardaremos la lista de todas las personas pertenecientes al instituto (tanto alumnos como profesores).

En esta clase podemos tener una única lista donde guardaríamos todas las personas del centro.

Pero otra opción que es la que yo haría es tener **tres listas**. Una para todas las personas, otra para los alumnos y otra para los profesores. De esta forma cuando introduzca por ejemplo un alumno, añado el objeto en la lista de personas y en la lista de alumnos.

Si en un momento dado quiero **mostrar todas las personas** del centro u obtener un listado con todos los teléfonos, lo hago sobre la lista de personas.

Si quiero buscar los **datos de un alumno**, lo hago sobre la lista de alumnos.

Si quiero ordenar los alumnos lo hago sobre la lista de alumnos sin necesidad de tocar la lista de personas.

Podría hacer algo así para empezar la clase:

```
public class ListaPersonas
{
    private List<Persona> mListaPersonas;
    private List<Persona> mListaAlumnos;
    private List<Persona> mListaProfesores;

    public ListaPersonas()
    {
        mListaPersonas = new List<Persona>();
        mListaAlumnos = new List<Persona>();
        mListaProfesores= new List<Persona>();
    }

    public void AnadirAlumno(string nombre, string dni, string tlf, string codCurso)
    {
        Alumno alumno = new Alumno(nombre, dni, tlf, codCurso);

        mListaPersonas.Add(alumno);
        mListaAlumnos.Add(alumno);
    }

    public void AnadirAlumno(Alumno alumno)
    {
        mListaPersonas.Add(alumno);
        mListaAlumnos.Add(alumno);
    }
}
```

Tenemos una **sobrecarga** de la función **AnadirAlumno**. Podemos añadir un alumno de dos formas distintas.

Una sería **pasando los datos** necesarios para un alumno y creando el objeto dentro del método.

La otra sería **creando el objeto fuera** (por ejemplo, en el formulario) y pasando el objeto para que lo añada a las listas.

Podeís hacerlo como queráis.

Tened en cuenta que en ambos métodos el **mismo objeto** se introduce en ambas listas. Si nosotros manipulamos un objeto en la lista de alumnos (por ejemplo, le añadimos una nota), afecta al de la lista de personas, porque son el mismo.