

## **Trabajo Practico N°3 - Laboratorio III**

### **HILOS**

Codifique un programa que contenga un método llamado "ejecutarHilo()" que imprima por consola la leyenda "HILO EJECUTADO" implemente el código necesario para que por medio de la ejecución de Hilos el método "ejecutarHilo()" se ejecute cada 30 segundos.

### **OBJETOS**

Escriba la definición de la clase CuentaBancaria cuyo alcance debe ser público:  
Tipos:

```
public string titularCta;  
public string cuit;  
public decimal saldo;  
public int numeroCBU;
```

Cree 2 instancias de la clase CuentaBancaria con los siguientes valores, respetando la definición de creación de la clase del punto anterior:

Instancia 1:  
titularCta: "Juan Alonso"  
cuit: "20-26987456-7".  
saldo: 1258.75.  
numeroCBU: 1236547896554

Instancia 2:  
titularCta: "Alberto Lopez"  
cuit: "23-15654321-9"  
saldo: 25698.78  
numeroCBU: 9876546546557

### **ENCAPSULAMIENTO**

Escriba la definición de la clase CuentaBancaria cuyo alcance debe ser público:  
Tipos:

```
private string titularCta;  
private string cuit;  
private decimal saldo;  
private int numeroCBU;
```

Codifique los métodos de encapsulamiento para los atributos de la clase y cree 2 instancias de la clase CuentaBancaria con los siguientes valores, respetando la definición de creación de la clase del punto anterior:

Instancia 1:

titularCta: "Juan Alonso"  
cuit: "20-26987456-7".  
saldo: 1258.75.  
numeroCBU: 1236547896554

Instancia 2:  
titularCta: "Alberto Lopez"  
cuit: "23-15654321-9"  
saldo: 25698.78  
numeroCBU: 9876546546557

## **HERENCIA Y POLIMORFISMO**

En primer lugar, cree una clase base llamada Figura y luego cree clases derivadas como Rectángulo, Circulo y Triangulo.

Incluya en la clase Figura un método virtual llamado Dibujar() que imprima por consola el mensaje "Dibuja Figura" y sobrescriba el método (override) en cada clase derivada para dibujar la forma determinada que representa la clase, es decir que para Rectángulo el método debe imprimir por consola "Dibuja Rectángulo", para Circulo "Dibuja Circulo" y para Triangulo "Dibuja Triangulo".

Finalmente en el método `static void Main()` cree un objeto `List<Figura>` y agregue al mismo instancias de Circulo, Triangulo y Rectángulo.

Utilice un bucle `foreach` para recorrer en iteración la lista y llamar al método Dibujar () en cada objeto Figura de la lista.

Aunque cada objeto de la lista tiene un tipo declarado de Figura, es el tipo en tiempo de ejecución el que se invocará.

El resultado del programa debería ser:

```
Dibuja Rectángulo
Dibuja Figura
Dibuja Círculo
Dibuja Figura
Dibuja Triangulo
Dibuja Figura
```

## **CLASES PARCIALES**

Cree un proyecto de Consola con el nombre: ClaseParcial

Agregue un archivo de clase llamado "Archivo1.cs".

En este archivo codifique la clase "partial class Rectángulo" que define las propiedades y atributos en este caso definir los atributos enteros ancho y alto y sus correspondientes propiedades.

En forma similar seguimos los pasos para crear el archivo de clase llamado Archivo2.cs y codificar la clase "partial class Rectángulo" que define dos métodos:

RetornarSuperficie() que será igual a  $\text{ancho} * \text{alto}$  y el método  
RetornarPerimetro() que será igual a  $\text{ancho} * 2 + \text{alto} * 2$

Finalmente codificamos la clase principal en el archivo Program.cs  
En la main creamos un objeto de la clase Rectángulo e inicializamos las propiedades y llamamos a sus métodos.

Para probarlo asigne al alto el valor de 5 y al ancho el valor de 10.