

## Ejercicio #1:

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

0 referencias
internal class InventarioTienda
{
    0 referencias
    public static void Main()
    {
        List<Tuple<int, string, int, decimal>> inventario = crearInventario();
        GestionarInventario(inventario);
    }

    1 referencia
    public static List<Tuple<int, string, int, decimal>> crearInventario()
    {
        return new List<Tuple<int, string, int, decimal>>();
    }

    1 referencia
    public static void GestionarInventario(List<Tuple<int, string, int, decimal>> inventario)
    {
        int opcion;
        do
```

```
1 referencia
    public static void GestionarInventario(List<Tuple<int, string, int, decimal>> inventario)
    {
        int opcion;
        do
        {
            Console.WriteLine("\n--- Menú de Inventario ---");
            Console.WriteLine("1. Agregar producto");
            Console.WriteLine("2. Eliminar producto");
            Console.WriteLine("3. Modificar producto");
            Console.WriteLine("4. Consultar producto");
            Console.WriteLine("5. Mostrar todos los productos");
            Console.WriteLine("6. Salir");
            Console.Write("Seleccione una opción: ");
            opcion = int.Parse(Console.ReadLine());

            switch (opcion)
            {
                case 1:
                    AgregarProducto(inventario);
                    break;
                case 2:
                    EliminarProducto(inventario);
                    break;
                case 3:
                    ModificarProducto(inventario);
                    break;
                case 4:
```

```

        break;
    case 4:
        ConsultarProducto(inventario);
        break;
    case 5:
        MostrarTodos(inventario);
        break;
    }
} while (opcion != 6);
}

```

1 referencia

```

public static void AgregarProducto(List<Tuple<int, string, int, decimal>> inventario)
{
    Console.Write("Código del producto: ");
    int codigo = int.Parse(Console.ReadLine());

    Console.Write("Nombre del producto: ");
    string nombre = Console.ReadLine();

    Console.Write("Cantidad: ");
    int cantidad = int.Parse(Console.ReadLine());

    Console.Write("Precio: ");
    decimal precio = decimal.Parse(Console.ReadLine());

    inventario.Add(new Tuple<int, string, int, decimal>(codigo, nombre, cantidad, precio));
    Console.WriteLine("Producto agregado.");
}

```

1 referencia

```

public static void EliminarProducto(List<Tuple<int, string, int, decimal>> inventario)
{
    Console.Write("Ingrese el código del producto a eliminar: ");
    int codigo = int.Parse(Console.ReadLine());

    var producto = inventario.Find(p => p.Item1 == codigo);

    if (producto != null)
    {
        inventario.Remove(producto);
        Console.WriteLine("Producto eliminado.");
    }
    else
    {
        Console.WriteLine("Producto no encontrado.");
    }
}

```

1 referencia

```
public static void ModificarProducto(List<Tuple<int, string, int, decimal>> inventario)
{
    Console.Write("Ingrese el código del producto a modificar: ");
    int codigo = int.Parse(Console.ReadLine());

    var producto = inventario.Find(p => p.Item1 == codigo);

    if (producto != null)
    {
        inventario.Remove(producto);

        Console.Write("Nuevo nombre: ");
        string nuevoNombre = Console.ReadLine();

        Console.Write("Nueva cantidad: ");
        int nuevaCantidad = int.Parse(Console.ReadLine());

        Console.Write("Nuevo precio: ");
        decimal nuevoPrecio = decimal.Parse(Console.ReadLine());

        inventario.Add(new Tuple<int, string, int, decimal>(codigo, nuevoNombre, nuevaCantidad, nuevoPrecio));
        Console.WriteLine("Producto modificado.");
    }
    else
    {
        Console.WriteLine("Producto no encontrado.");
    }
}
```

1 referencia

```
public static void ConsultarProducto(List<Tuple<int, string, int, decimal>> inventario)
{
    Console.Write("Ingrese el código del producto: ");
    int codigo = int.Parse(Console.ReadLine());

    var producto = inventario.Find(p => p.Item1 == codigo);

    if (producto != null)
    {
        Console.WriteLine($"Código: {producto.Item1}, Nombre: {producto.Item2}, Cantidad: {producto.Item3}, Precio: {producto.Item4}");
    }
    else
    {
        Console.WriteLine("Producto no encontrado.");
    }
}
```

1 referencia

```
public static void MostrarTodos(List<Tuple<int, string, int, decimal>> inventario)
{
    foreach (var producto in inventario)
    {
        Console.WriteLine($"Código: {producto.Item1}, Nombre: {producto.Item2}, Cantidad: {producto.Item3}, Precio: {producto.Item4}");
    }
}
```

--- Menú de Inventario ---

1. Agregar producto
2. Eliminar producto
3. Modificar producto
4. Consultar producto
5. Mostrar todos los productos
6. Salir

Seleccione una opción: 1

Código del producto: 123

Nombre del producto: Cafe

Cantidad: 4

Precio: 10

Producto agregado.

--- Menú de Inventario ---

1. Agregar producto
2. Eliminar producto
3. Modificar producto
4. Consultar producto
5. Mostrar todos los productos
6. Salir

Seleccione una opción: 3

Ingrese el código del producto a modificar: 123

Nuevo nombre: cafe

Nueva cantidad: 7

Nuevo precio: 8

Producto modificado.

--- Menú de Inventario ---

1. Agregar producto
2. Eliminar producto
3. Modificar producto
4. Consultar producto
5. Mostrar todos los productos
6. Salir

Seleccione una opción: 4

Ingrese el código del producto: 123

Código: 123, Nombre: cafe, Cantidad: 7, Precio: 8

--- Menú de Inventario ---

1. Agregar producto
2. Eliminar producto
3. Modificar producto
4. Consultar producto
5. Mostrar todos los productos
6. Salir

Seleccione una opción: 5

Código: 123, Nombre: cafe, Cantidad: 7, Precio: 8

--- Menú de Inventario ---

1. Agregar producto
2. Eliminar producto
3. Modificar producto
4. Consultar producto
5. Mostrar todos los productos
6. Salir

Seleccione una opción: 2

6. Salir

Seleccione una opción: 2

Ingrese el código del producto a eliminar: 123

Producto eliminado.

--- Menú de Inventario ---

1. Agregar producto
2. Eliminar producto
3. Modificar producto
4. Consultar producto
5. Mostrar todos los productos
6. Salir

Seleccione una opción: 5

--- Menú de Inventario ---

1. Agregar producto
2. Eliminar producto
3. Modificar producto
4. Consultar producto
5. Mostrar todos los productos
6. Salir

Seleccione una opción: 4

Ingrese el código del producto: 123

Producto no encontrado.

--- Menú de Inventario ---

1. Agregar producto
2. Eliminar producto
3. Modificar producto
4. Consultar producto

```
4. Consultar producto
5. Mostrar todos los productos
6. Salir
Seleccione una opción: 4
Ingrese el código del producto: 123
Producto no encontrado.
```

```
--- Menú de Inventario ---
1. Agregar producto
2. Eliminar producto
3. Modificar producto
4. Consultar producto
5. Mostrar todos los productos
6. Salir
Seleccione una opción: 6
```

```
C:\Users\angel\OneDrive\Desktop\Metodología_y_programaci
ercicio1.exe (proceso 3756) se cerró con el código 0 (0x
Para cerrar automáticamente la consola cuando se detiene
```

### Ejercicio #2:

```

using System;
using System.Collections.Generic;

0 referencias
class Program
{
    0 referencias
    static void Main()
    {
        List<decimal> transacciones = new List<decimal>();
        bool salir = false;

        while (!salir)
        {
            Console.WriteLine("Menú de opciones:");
            Console.WriteLine("1. Consultar saldo");
            Console.WriteLine("2. Depositar dinero");
            Console.WriteLine("3. Retirar dinero");
            Console.WriteLine("4. Salir");
            Console.Write("Elige una opción: ");
            string opcion = Console.ReadLine();

            switch (opcion)
            {
                case "1":
                    decimal saldo = CalcularSaldo(transacciones);
                    Console.WriteLine($"Saldo actual: {saldo:C}");
                    break;
            }
        }
    }
}

```



```

        case "2":
            Console.Write("Ingresa la cantidad a depositar: ");
            decimal deposito = Convert.ToDecimal(Console.ReadLine());
            transacciones.Add(deposito);
            Console.WriteLine("Depósito realizado.");
            break;

        case "3":
            Console.Write("Ingresa la cantidad a retirar: ");
            decimal retiro = Convert.ToDecimal(Console.ReadLine());
            decimal saldoActual = CalcularSaldo(transacciones);
            if (retiro <= saldoActual)
            {
                transacciones.Add(-retiro);
                Console.WriteLine("Retiro realizado.");
            }
            else
            {
                Console.WriteLine("Saldo insuficiente.");
            }
            break;

        case "4":
            salir = true;
            Console.WriteLine("Saliendo del programa.");
            break;

        default:
            Console.WriteLine("Opción no válida.");
            break;
    }
    Console.WriteLine();
}

```

2 referencias

```

static decimal CalcularSaldo(List<decimal> transacciones)
{
    decimal saldo = 0;
    foreach (var transaccion in transacciones)
    {
        saldo += transaccion;
    }
    return saldo;
}

```

Menú de opciones:

1. Consultar saldo
2. Depositar dinero
3. Retirar dinero
4. Salir

Elige una opción: 1

Saldo actual: C\$0.00

Menú de opciones:

1. Consultar saldo
2. Depositar dinero
3. Retirar dinero
4. Salir

Elige una opción: 2

Ingresa la cantidad a depositar: 4551

Depósito realizado.

Menú de opciones:

1. Consultar saldo
2. Depositar dinero
3. Retirar dinero
4. Salir

Elige una opción: 1

Saldo actual: C\$4,551.00

Menú de opciones:

1. Consultar saldo
2. Depositar dinero
3. Retirar dinero
4. Salir

4. Salir

Elige una opción: 1

Saldo actual: C\$4,551.00

Menú de opciones:

1. Consultar saldo

2. Depositar dinero

3. Retirar dinero

4. Salir

Elige una opción: 3

Ingresar la cantidad a retirar: 8444

Saldo insuficiente.

Menú de opciones:

1. Consultar saldo

2. Depositar dinero

3. Retirar dinero

4. Salir

Elige una opción: 3

Ingresar la cantidad a retirar: 800

Retiro realizado.

Menú de opciones:

1. Consultar saldo

2. Depositar dinero

3. Retirar dinero

4. Salir

Elige una opción: 1

Saldo actual: C\$3,751.00

```
2. Depositar dinero
3. Retirar dinero
4. Salir
Elige una opción: 1
Saldo actual: C$3,751.00
```

```
Menú de opciones:
1. Consultar saldo
2. Depositar dinero
3. Retirar dinero
4. Salir
Elige una opción: 4
Saliendo del programa.
```

```
C:\Users\angel\source\repos\Ejerc
código 0 (0x0).
```

### Ejercicio #3:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

internal class DiccionarioT
{
    public static void Main()
    {
        List<Tuple<string, string>> diccionario = crearDiccionario();
        traducir(diccionario);
    }

    public static List<Tuple<string, string>> crearDiccionario()
    {
        List<Tuple<string, string>> diccionario = new List<Tuple<string, string>>();
        for (int i = 0; i < 5; i++)
        {
            Console.WriteLine($"Introduce la palabra {i+1} en inglés: ");
            string palabra1 = Console.ReadLine();
            Console.WriteLine($"Introduce la palabra {i + 1} en español: ");
            string palabra2 = Console.ReadLine();

            Console.WriteLine($"Introduce la palabra {i + 1} en español: ");
            string palabra2 = Console.ReadLine();

            diccionario.Add(new Tuple<string, string>(palabra1, palabra2));
        }
        return diccionario;
    }

    public static void traducir (List<Tuple<string,string>> diccionario)
    {
        Console.Write("Introduzca la palabra a traducir: ");
        string palabraTrad = Console.ReadLine();
        bool encontrado = false;

        foreach (var duo in diccionario)
        {
            if (duo.Item1.Equals(palabraTrad, StringComparison.OrdinalIgnoreCase))
            {
                Console.WriteLine($"La traducción de {palabraTrad} es : {duo.Item2}");
                encontrado = true;
                break;
            }
            else if (duo.Item2.Equals(palabraTrad, StringComparison.OrdinalIgnoreCase))
            {
                Console.WriteLine($"La traducción de {palabraTrad} es : {duo.Item1}");
                encontrado = true;
                break;
            }
        }
    }
}
```

```

        {
            Console.WriteLine($"La traducción de {palabraTrad} es : {duo.Item1}");
            encontrado = true;
            break;
        }
    }
    if (!encontrado)
    {
        Console.WriteLine("La palabra no se encuentra en el diccionario");
    }
}

```

Consola de depuración de Mi: x + v

```

Introduce la palabra 1 en inglés:
Hello
Introduce la palabra 1 en español:
Hola
Introduce la palabra 2 en inglés:
Cowboy
Introduce la palabra 2 en español:
Vaquero
Introduce la palabra 3 en inglés:
Apple
Introduce la palabra 3 en español:
Manzana
Introduce la palabra 4 en inglés:
Tree
Introduce la palabra 4 en español:
Árbol
Introduce la palabra 5 en inglés:
Name
Introduce la palabra 5 en español:
Nombre
Introduzca la palabra a traducir: hola
La traducción de hola es : Hello

C:\Users\angel\OneDrive\Desktop\Metodología_y_programacion_estructurada\DiccionarioT\DiccionarioT\DiccionarioT\bin\
\net8.0\DiccionarioT.exe (proceso 4144) se cerró con el código 0 (0x0).
Para cerrar automáticamente la consola cuando se detiene la depuración, habilite Herramientas ->Opciones ->Depura
Cerrar la consola automáticamente al detenerse la depuración.
Presione cualquier tecla para cerrar esta ventana. . .

```