

TP2 Laboratorio 2, Cardullo Matias

Hice un sistema para la gestión de una librería.

El formulario principal muestra cuanto dinero tenemos en el cajero, cuantos libros tenemos, cuanto valen en general y por tipo, y la información exacta de cada uno por separado.

Libreria Cardullo

Cajero: \$0

Capacidad de la biblioteca: 3000
Total por Manuales: \$159,11
Total por Novelas: \$364,88
Valor Total: \$523,99

Listado de Libros

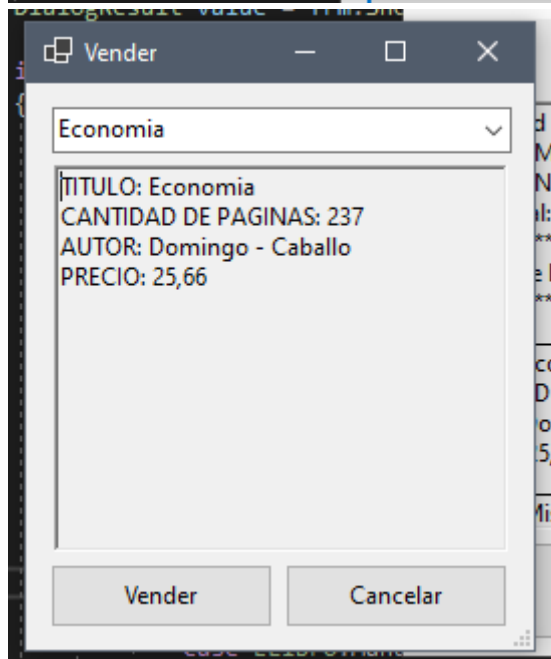
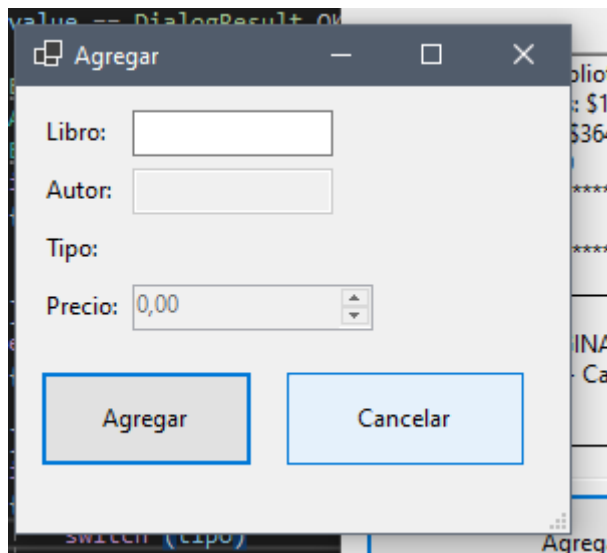
TITULO: Economia
CANTIDAD DE PAGINAS: 237
AUTOR: Domingo - Caballo
PRECIO: 25,66

TITULO: Miseria

Agregar Libro

Vender Libro

Despues tenemos dos botones, uno para agregar libro y otro para vender libro, cada uno abre un respectivo form.



Temas Utilizados:

Clase 4: Sobrecarga de operadores en la clase librería;

```

1 referencia
public static bool operator ==(Libreria e, Libro l)
{
    bool retorno = false;
    foreach (Libro item in e._libros)
    {
        if(item == l)
        {
            retorno = true;
            break;
        }
    }
    return retorno;
}

0 referencias
public static bool operator !=(Libreria e, Libro l)
{
    return !(e == l);
}

8 referencias
public static Libreria operator +(Libreria e, Libro l)
{
    if(e._libros.Count < e._capacidad)
    {
        foreach (Libro item in e._libros) [...]
        e._libros.Add(l);
    }
    return e;
}

1 referencia
public static Libreria operator -(Libreria e, int id)
{

```

Clase 6: Se implementaron 3 Windows Form

Clase 7: Use una colección del tipo List<Libros> para el registro de los libros.

```

private int _capacidad;
private List<Libro> _libros;

```

Clase 8: Propiedades y encapsulamiento para la clase librería.

```

2referencias
public int Capacidad
{
    get
    {
        return _capacidad;
    }
}

1referencia
public double PrecioDeManuales
{
    get
    {
        return ObtenerPrecio(ELibro.Manual);
    }
}

1referencia
public double PrecioDeNovelas
{
    get
    {
        return ObtenerPrecio(ELibro.Novela);
    }
}

1referencia
public double PrecioTotal
{
    get
    {
        return ObtenerPrecio(ELibro.Novela)+ ObtenerPrecio(ELibro.Manual);
    }
}

2referencias
public List<Libro> Libros
{
    get { return _libros; }
    set { _libros = value; }
}

1referencia

```

Clase 9, 10 y 11: Polimorfismo ; la clase Libro es abstracta y las clases Novela y Manual heredan de Libro

20referencias

```

public class Novela : Libro
{

```

```

public class Manual : Libro
{
    public ETipo tipo;

```

30referencias

```

public abstract class Libro
{

```