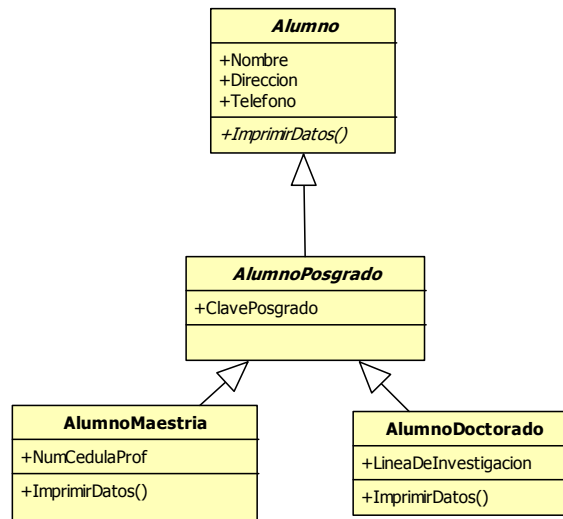


EJERCICIOS – POO CON C#

1. Realizar el diagrama de clase, la codificación en C# y el programa que utilice la clase para un "Usuario" con dos atributos: LoginCorrecto y PasswordCorrecto y un método: Validar(login, pass) : bool, que se encargará de comparar los parámetros suministrados con los valores correctos. El valor de retorno será "true" si son iguales y "false" si son diferentes.
2. Realice el diagrama de clases y la codificación en C# de las siguientes clases utilizando composición:
Un "Alumno" con Nombre, Grado y Grupo "Tiene una" "Nota Final" con Calificación y ComentarioDelProfesor.
Realice un programa que utilice la clase.
3. Utilizando herencia y clases abstractas, codifique en C# una jerarquia de clases para el siguiente ejemplo:



4. Realice una clase en C# para una Cuenta Bancaria y el programa correspondiente que utilice la clase.
Tomar en cuenta que:
 - La cuenta tiene un número de identificación que debe ser suministrado únicamente al momento de crear un nuevo objeto. Y puede ser solamente leído al utilizar la clase.
 - La cuenta tiene un saldo.
 - El saldo puede consultarse, y solo se puede modificar dentro de la clase.
 - La única manera de modificar el saldo externamente es Depositando una cantidad o Retirando una Cantidad.
 - Al depositar, el saldo se incrementa. Al retirar, el saldo decrementa. En ambos casos se notifica el depósito o retiro con un mensaje.
 - Solo se puede retirar una cantidad menor o igual al saldo actual.
 - La clase "cuenta" tiene un método para Obtener el saldo con formato moneda (regresa un string) y un metodo para imprimir el saldo actual.
5. Modifique la clase anterior para que sea abstracta. La operación "RealizarRetiro" será virtual.
Dos clases heredarán de esta clase abstracta:
 - CuentaBancariaSinComisiones (Que toma todos los elementos heredados tal cual están).
 - CuentaBancariaConComisiones (Que sobre-escribe la operación virtual "Retirar", y tiene una propiedad adicional: Comisión que es una cantidad que se cobra al realizar cada retiro).
 - Al retirar una cantidad, en realidad el saldo se modifica de la siguiente manera:
Saldo = Saldo – cantidad – Comisión
 - Si el saldo no es suficiente para cargar ambos (cantidad y Comisión), no se realiza el retiro y se notifica con un mensaje.Tanto CuentaBancariaSinComisiones como CuentaBancariaConComisiones requieren que se indique el Número de cuenta al momento de instanciarse.

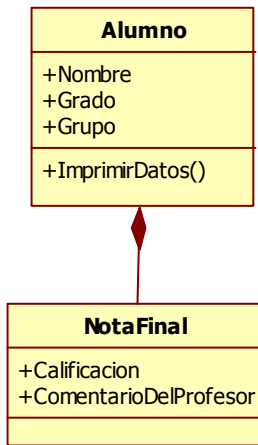
SOLUCION PROBLEMA #1

Usuario
+LoginCorrecto +PasswordCorrecto
+Validar(string login, string pass): bool

```
class Usuario
{
    private string _loginCorrecto;
    private string _passwordCorrecto;
    public string LoginCorrecto
    {
        get { return _loginCorrecto; }
        set { _loginCorrecto = value; }
    }
    public string PasswordCorrecto
    {
        get { return _passwordCorrecto; }
        set { _passwordCorrecto = value; }
    }
    public bool Validar(string log, string pass)
    {
        if (log == LoginCorrecto && pass == PasswordCorrecto)
        {
            return true;
        }
        else
        {
            return false;
        }
    }
}

class Programa
{
    static void Main()
    {
        Usuario miUsuario = new Usuario();
        miUsuario.LoginCorrecto = "Administrador";
        miUsuario.PasswordCorrecto = "Admin";
        if (miUsuario.Validar ( "Administrador", "Admin"))
        {
            System.Console.WriteLine(" .... BIENVENIDO .... ");
        }
        else
        {
            System.Console.WriteLine(" .....ACCESO NEGADO.....");
        }
        System.Console.ReadLine();
    }
}
```

SOLUCION PROBLEMA #2



```

class NotaFinal
{
    private double _calif;
    private string _comentario;
    public double Calificacion
    {
        get { return _calif; }
        set { _calif = value; }
    }
    public string Comentario
    {
        get { return _comentario; }
        set { _comentario = value; }
    }
}

class Alumno
{
    private string _nombre;
    private int _grado;
    private string _grupo;
    private NotaFinal _notaFinal = new NotaFinal ();
    public string Nombre
    {
        get { return _nombre; }
        set { _nombre = value; }
    }
    public int Grado
    {
        get { return _grado; }
        set { _grado = value; }
    }
    public string Grupo
    {
        get { return _grupo; }
        set { _grupo = value; }
    }
    public NotaFinal Nota
    {
        get { return _notaFinal; }
        set { _notaFinal = value; }
    }
    public void ImprimirDatos()
    {
        System.Console.WriteLine(" Datos del alumno : " + Nombre);
        System.Console.WriteLine(" Calificacion : " + Nota.Calificacion);
        System.Console.WriteLine(" Comentario : " + Nota.Comentario);
    }
}

class Programa
{
    static void Main()
    {
        Alumno a = new Alumno();
        a.Nombre = "Juan";
        a.Grado = 2;
        a.Grupo = "A";
        a.Nota.Calificacion = 100;
        a.Nota.Comentario = "Excelente desempeño";
        a.ImprimirDatos();
        System.Console.ReadLine();
    }
}
  
```

SOLUCION PROBLEMA #3

```

abstract class Alumno
{
    private string _nombre;
    private string _direccion;
    private string _telefono;
    public string Nombre
    {
        get { return _nombre; }
        set { _nombre = value; }
    }
    public string Direccion
    {
        get { return _direccion; }
        set { _direccion = value; }
    }
    public string Telefono
    {
        get { return _telefono; }
        set { _telefono = value; }
    }
    public abstract void ImprimirDatos();
}
abstract class AlumnoPosgrado : Alumno
{
    private string _clavePosgrado;
    public string ClavePosgrado
    {
        get { return _clavePosgrado; }
        set { _clavePosgrado = value; }
    }
}
class AlumnoMaestria : AlumnoPosgrado
{
    private string _numCedulaProfesional;
    public string NumeroCedulaProfesional
    {
        get { return _numCedulaProfesional; }
        set { _numCedulaProfesional = value; }
    }
    override public void ImprimirDatos()
    {
        System.Console.WriteLine(" Datos del alumno de Maestria ");
        System.Console.WriteLine(" Nombre: " + Nombre +
                                   " Direccion: " + Direccion +
                                   " Telefono: " + Telefono +
                                   " ClavePosgrado: " + ClavePosgrado +
                                   " Num.Ced.Profesional" + NumeroCedulaProfesional);
    }
}
class AlumnoDoctorado : AlumnoPosgrado
{
    private string _lineaDeInvestigacion;
    public string LineaDeInvestigacion
    {
        get { return _lineaDeInvestigacion; }
        set { _lineaDeInvestigacion = value; }
    }
    override public void ImprimirDatos()
    {
        System.Console.WriteLine(" Datos del alumno de Doctorado ");
        System.Console.WriteLine(" Nombre: " + Nombre +
                                   " Direccion: " + Direccion +
                                   " Telefono: " + Telefono +
                                   " ClavePosgrado: " + ClavePosgrado +
                                   " Linea de Investigacion" + LineaDeInvestigacion );
    }
}

class Programa
{
    static void Main()
    {
        AlumnoMaestria am = new AlumnoMaestria();
        AlumnoDoctorado ad = new AlumnoDoctorado();
        am.Nombre = "Juan";
        am.Direccion = " Guerrero 1111";
        am.Telefono = "711-11-11";
        am.ClavePosgrado = "POSGRADO01";
        am.NumeroCedulaProfesional = "XXYZ2323";
        am.ImprimirDatos();

        ad.Nombre = "Pedro";
        ad.Direccion = " Juarez 3434";
        ad.Telefono = " 716-16-16";
        ad.ClavePosgrado = "POSGRADO02";
        ad.LineaDeInvestigacion = "POO";
        ad.ImprimirDatos();
        System.Console.ReadLine();
    }
}

```

SOLUCION PROBLEMA #4

```
class CuentaBancaria
{
    private double _saldo = 0;
    private int _numeroDeCuenta;
    public CuentaBancaria(int numeroDeCuenta)
    {
        _numeroDeCuenta = numeroDeCuenta;
    }
    public int NumeroDeCuenta
    {
        get { return _numeroDeCuenta; }
    }
    public double Saldo
    {
        get { return _saldo; }
        private set { _saldo = value; }
    }
    public void RealizarDeposito(double cantidad)
    {
        Saldo = Saldo + cantidad;
        System.Console.WriteLine(" Se han depositado {0} en la cuenta numero {1} ",
                                cantidad, NumeroDeCuenta);
    }
    public void RealizarRetiro(double cantidad)
    {
        if (cantidad <= Saldo)
        {
            Saldo = Saldo - cantidad;
            System.Console.WriteLine(" Se han retirado {0} de la cuenta numero {1} ",
                                    cantidad, NumeroDeCuenta);
        }
    }
    public string ObtenerSaldoConFormato()
    {
        return Saldo.ToString("c2");
    }
    public void ImprimirSaldo()
    {
        System.Console.WriteLine(" El saldo en la cuenta {0} es de {1} ", NumeroDeCuenta, Saldo);
    }
}

class Programa
{
    static void Main()
    {
        CuentaBancaria miCuenta = new CuentaBancaria(2563123);
        miCuenta.RealizarDeposito(500.00);
        miCuenta.ImprimirSaldo();
        miCuenta.RealizarRetiro(100.00);
        miCuenta.ImprimirSaldo();
        miCuenta.RealizarDeposito(300.00);
        miCuenta.ImprimirSaldo();
        miCuenta.RealizarRetiro(200.00);
        miCuenta.ImprimirSaldo();

        System.Console.ReadLine();
    }
}
```

SOLUCION PROBLEMA #5

```

abstract class CuentaBancaria
{
    private double _saldo = 0;
    private int _numeroDeCuenta;
    public CuentaBancaria(int numeroDeCuenta)
    {
        _numeroDeCuenta = numeroDeCuenta;
    }
    public int NumeroDeCuenta
    {
        get { return _numeroDeCuenta; }
    }
    public double Saldo
    {
        get { return _saldo; }
        protected set { _saldo = value; }
    }
    public void RealizarDeposito(double cantidad)
    {
        Saldo = Saldo + cantidad;
        System.Console.WriteLine(" Se han depositado {0} en la cuenta numero {1} ",
            cantidad, NumeroDeCuenta);
    }
    public virtual void RealizarRetiro(double cantidad)
    {
        if (cantidad <= Saldo)
        {
            Saldo = Saldo - cantidad;
            System.Console.WriteLine(" Se han retirado {0} en la cuenta numero {1} ",
                cantidad, NumeroDeCuenta);
        }
    }
    public string ObtenerSaldoConFormato()
    {
        return Saldo.ToString("c2");
    }
    public void ImprimirSaldo()
    {
        System.Console.WriteLine(" El saldo en la cuenta {0} es de {1} ",
            NumeroDeCuenta, ObtenerSaldoConFormato());
    }
}
//*****En esta cuenta bancaria no se cobra ninguna comision
class CuentaBancariaSinComisiones : CuentaBancaria
{
    public CuentaBancariaSinComisiones(int numeroDeCuenta) : base(numeroDeCuenta)
    { }
}
//*****En esta cuenta bancaria se cobra una comision por cada retiro
class CuentaBancariaConComisiones : CuentaBancaria
{
    private double _comision;
    public double Comision
    {
        get { return _comision; }
        set { _comision = value; }
    }
    public CuentaBancariaConComisiones(int numeroDeCuenta) : base(numeroDeCuenta)
    { }
    public override void RealizarRetiro(double cantidad)
    {
        if ((cantidad + Comision) <= Saldo)
        {
            Saldo = Saldo - cantidad - Comision;
            System.Console.WriteLine(" Se han retirado: " + cantidad.ToString("c2"));
            System.Console.WriteLine(" Se han retirado: " + Comision.ToString("c2") +
                " por comision");
        }
        else
        {
            System.Console.WriteLine(" Saldo insuficiente para hacer el retiro + la comision ");
        }
    }
}
class Programa
{
    static void Main()
    {
        CuentaBancariaConComisiones miCuenta = new CuentaBancariaConComisiones(234567);
        miCuenta.Comision = 20;
        miCuenta.ImprimirSaldo();
        miCuenta.RealizarDeposito(100.00);
        miCuenta.ImprimirSaldo();
        miCuenta.RealizarRetiro(10.00);
        miCuenta.ImprimirSaldo();

        CuentaBancariaSinComisiones miCuenta2 = new CuentaBancariaSinComisiones(9876543);
        miCuenta2.ImprimirSaldo();
        miCuenta2.RealizarDeposito(100.00);
        miCuenta2.ImprimirSaldo();
        miCuenta2.RealizarRetiro(10.00);
        miCuenta2.ImprimirSaldo();

        System.Console.ReadLine();
    }
}

```