

Clases e interfaces del proyecto.

Fecha

```
package uam.edoo.laboratorioUno.clases;

/**
 * @author Equipo de trabajo
 *
 */
public class Fecha
{

    private int mes;
    private int annio;
    /**
     * @return the mes
     */
    public int getMes()
    {
        return mes;
    }
    /**
     * @param mes the mes to set
     */
    public void setMes( int mes )
    {
        this.mes = mes;
    }
    /**
     * @return the annio
     */
    public int getAnnio()
    {
        return annio;
    }
    /**
     * @param annio the annio to set
     */
    public void setAnnio( int annio )
    {
        this.annio = annio;
    }

}
```

Tarjeta

```
package uam.edoo.laboratorioUno.clases;

/**
 * @author Equipo de trabajo
 *
 */
public class Tarjeta
{

    private String nombreTarjeta;
    private Fecha fechaVencimiento;
    private String numeroTarjeta;
    private String numeroCuenta;
    private String tipo;
    private float saldo;
    /**
     * @return the nombreTarjeta
     */
    public String getNombreTarjeta()
    {
        return nombreTarjeta;
    }
    /**
     * @param nombreTarjeta the nombreTarjeta to set
     */
    public void setNombreTarjeta( String nombreTarjeta )
    {
        this.nombreTarjeta = nombreTarjeta;
    }
    /**
     * @return the fechaVencimiento
     */
    public Fecha getFechaVencimiento()
    {
        return fechaVencimiento;
    }
    /**
     * @param fechaVencimiento the fechaVencimiento to set
     */
    public void setFechaVencimiento( Fecha fechaVencimiento )
    {
        this.fechaVencimiento = fechaVencimiento;
    }
    /**
     * @return the numeroTarjeta
     */
    public String getNumeroTarjeta()
    {
        return numeroTarjeta;
    }
    /**
     * @param numeroTarjeta the numeroTarjeta to set
     */
    public void setNumeroTarjeta( String numeroTarjeta )
    {

```

```

    this.numeroTarjeta = numeroTarjeta;
}
/**
 * @return the numeroCuenta
 */
public String getNumeroCuenta()
{
    return numeroCuenta;
}
/**
 * @param numeroCuenta the numeroCuenta to set
 */
public void setNumeroCuenta( String numeroCuenta )
{
    this.numeroCuenta = numeroCuenta;
}
/**
 * @return the tipo
 */
public String getTipo()
{
    return tipo;
}
/**
 * @param tipo the tipo to set
 */
public void setTipo( String tipo )
{
    this.tipo = tipo;
}
/**
 * @return the saldo
 */
public float getSaldo()
{
    return saldo;
}
/**
 * @param saldo the saldo to set
 */
public void setSaldo( float saldo )
{
    this.saldo = saldo;
}

/**
 * Se sobre escribe el método toString para que regrese una cadena
 * (String) con la información que se desea se imprima en pantalla.
 * El símbolo + se utiliza para ir concatenando los valores a la cadena
 * a regresar, la fecha de vencimiento, al ser un objeto se imprime
 * a través del valor de sus atributos (mes y año)
 */
public String toString() {
    String mensaje = "";
    mensaje = "Titular: " + this.nombreTarjeta + "\n"+
              "Número de tarjeta: " + this.numeroTarjeta + "\n"+
              "Número de cuenta: " + this.numeroCuenta + "\n"+

```

```

        "Tipo: " + this.tipo + "\n"+
        "Fecha de vencimiento: " + fechaVencimiento.getMes() +
        "/" + fechaVencimiento.getAnio();
    return mensaje;
}
}

```

TarjetaCD

```

package uam.edoo.laboratorioUno.clases;

/**
 * @author Equipo de trabajo
 *
 */
public class TarjetaCD extends Tarjeta
{
    //Se agrega un atributo llamado género
    //exclusivo de una Tarjeta de CD
    private String genero;

    //Sobre escritura del método constructos
    //para que reciba el género de música
    public TarjetaCD(String genero) {
        this.genero = genero;
    }

    /**
     * @return the genero
     */
    public String getGenero()
    {
        return genero;
    }

    /**
     * @param genero the genero to set
     */
    public void setGenero( String genero )
    {
        this.genero = genero;
    }

    /**
     * Se sobre escribe el método to String
     * de la clase Tarjeta para que imprima
     * la misma información y agregue el
     * género
     */
    public String toString() {
        return super.toString() + "Genero: " + this.genero;
    }
}

```

TarjetaPeliculas

```
package uam.edoo.laboratorioUno.clases;

/**
 * @author Equipo de trabajo
 *
 */
public class TarjetaPeliculas
{
    private String formato;

    /**
     * @return the formato
     */
    public String getFormato()
    {
        return formato;
    }

    /**
     * @param formato the formato to set
     */
    public void setFormato( String formato )
    {
        this.formato = formato;
    }

    public String toString() {
        String mensaje = "";
        return super.toString() + "\n Consola: " + this.formato;
    }
}
```

TarjetaVideoJuegos

```
package uam.edoo.laboratorioUno.clases;

/**
 * @author Equipo de trabajo
 *
 */
public class TarjetaVideoJuegos extends Tarjeta
{
    private String consola;

    /**
     * @return the consola
     */
    public String getConsola()
    {
        return consola;
    }
}
```

```

/**
 * @param consola the consola to set
 */
public void setConsola( String consola )
{
    this.consola = consola;
}

public String toString() {
    String mensaje = "";
    return super.toString() + "\n Consola: " + this.consola;
}
}

```

OperacionesTarjeta

```

package uam.edoo.laboratorioUno.interfaces;

import uam.edoo.laboratorioUno.clases.Tarjeta;

/**
 * @author Equipo de trabajo
 */
public interface OperacionesTarjeta
{
    /**
     * Método que imprime los datos de cualquier tipo
     * de tarjeta
     * @param tarjeta
     */
    public void imprimirDatos(Tarjeta tarjeta);

    /**
     * Método que abona saldo a una tarjeta de cualquier tipo
     * y la regresa
     * @param tarjeta
     * @param saldo
     * @return
     */
    public Tarjeta abonarSaldo(Tarjeta tarjeta, float saldo);
}

```

LlenarDatos

```
package uam.edoo.laboratorioUno.operaciones;

import uam.edoo.laboratorioUno.clases.*;

import java.io.*;

/**
 * @author Equipo de trabajo
 *
 */
public class LlenarDatos
{
    /**
     * Método que regresa un objeto tipo Tarjeta con
     * sus atributos llenos.
     * @return
     */
    public Tarjeta llenarDatos() {

        Tarjeta tarjeta = new Tarjeta();
        Fecha fechaVencimiento = new Fecha();

        tarjeta.setNombreTarjeta( "Nombre del titular" );
        tarjeta.setNumeroCuenta( "123456" );
        tarjeta.setNumeroTarjeta( "123 456 789" );
        tarjeta.setTipo( "General" );
        //Para indicar un tipo flotante, se coloca la letra F después del
valor
        tarjeta.setSaldo( 100.50F );

        //Se llena el atributo mes del objeto Fecha
        fechaVencimiento.setMes( 11 );
        //Se llena el atributo año del objeto Fecha
        fechaVencimiento.setAño( 18 );

        //Se asigna el objeto al atributo tipo Fecha de la tarjeta creada
        tarjeta.setFechaVencimiento( fechaVencimiento );
        return tarjeta;
    }

    /**
     * Método que crea y llena un objeto tipo
     * Tarjeta a través de valores leídos desde
     * teclado.
     * @return
     */
    public Tarjeta desdeTeclado() {
        Tarjeta tarjeta = new Tarjeta();
        Fecha fechaVencimiento = new Fecha();

        //Variables que recibe el valor leído
        //desde teclado a partir del método
        //leerDato()
        String dato = "";
    }
}
```

```

String numeroTarjeta = "";

System.out.println("Nombre del cliente:");
dato = leerDato();
tarjeta.setNombreTarjeta(dato);

System.out.println("Numero de cuenta:");
dato = leerDato();
tarjeta.setNumeroCuenta(dato);

//En este caso la lectura se hace usando
//la variable numeroTarjeta
System.out.println("Numero de tarjeta:");
numeroTarjeta = leerDato();
tarjeta.setNumeroTarjeta(numeroTarjeta);

System.out.println("Tipo:");
dato = leerDato();
tarjeta.setTipo(dato);

//Cuando se leen número es necesario
//parsear (convertir) la cadena leida
//a su equivalente numérico, en este caso
//con la clase Float y su método parseFloat
System.out.println("Saldo de la tarjeta:");
dato = leerDato();
tarjeta.setSaldo(Float.parseFloat(dato));

//Para convertir cadenas a enteros, se utiliza
//Integer.parseInt
System.out.println("Mes de vencimiento:");
dato = leerDato();
fechaVencimiento.setMes(Integer.parseInt(dato));

System.out.println("Año de vencimiento:");
dato = leerDato();
fechaVencimiento.setAnio(Integer.parseInt(dato));

tarjeta.setFechaVencimiento(fechaVencimiento);

return tarjeta;
}
/**
 * Método que lee desde teclado y lo asigna a un String
 * para que sea regresado, se deben utilizar las clases
 * InputStreamReader y BufferedReader para realizar la
 * lectura
 * @return
 */
public String leerDato() {
    InputStreamReader input = new InputStreamReader(System.in);
    BufferedReader entrada = new BufferedReader(input);
    String datoLeido="";

    try {
        datoLeido = entrada.readLine();
    } catch (IOException e) {

```

```

        System.out.println("ERROR DESDE TECLADO");
        e.printStackTrace();
    }

    return datoLeido;
}

/**
 * Este método regresa una TarjetaCD con sus datos
 * llenos, recibe como parámetro el género que se
 * quiere asignar a la tarjeta
 * @return
 */
public TarjetaCD llenarDatosCD(String genero) {

    TarjetaCD tarjeta = new TarjetaCD(genero);
    Fecha fechaVencimiento = new Fecha();

    tarjeta.setNombreTarjeta( "Nombre del titular" );
    tarjeta.setNumeroCuenta( "123456" );
    tarjeta.setNumeroTarjeta( "123 456 789" );
    tarjeta.setTipo( "Tarjeta de CD" );
    //Para indicar un tipo flotante, se coloca la letra F después del
valor
    tarjeta.setSaldo( 100.50F );

    //Se llena el atributo mes del objeto Fecha
    fechaVencimiento.setMes( 11 );
    //Se llena el atributo año del objeto Fecha
    fechaVencimiento.setAño( 18 );

    //Se asigna el objeto al atributo tipo Fecha de la tarjeta creada
    tarjeta.setFechaVencimiento( fechaVencimiento );
    return tarjeta;
}

public TarjetaVideoJuegos llenarDatosVideoJuegos() {

    TarjetaVideoJuegos tarjeta = new TarjetaVideoJuegos();
    Fecha fechaVencimiento = new Fecha();

    tarjeta.setNombreTarjeta( "Nombre del titular" );
    tarjeta.setNumeroCuenta( "123456" );
    tarjeta.setNumeroTarjeta( "123 456 789" );
    tarjeta.setTipo( "Tarjeta de CD" );
    //Para indicar un tipo flotante, se coloca la letra F después del
valor
    tarjeta.setSaldo( 100.50F );

    //Se asigna una consola
    tarjeta.setConsola( "X-Box 360" );

    //Se llena el atributo mes del objeto Fecha
    fechaVencimiento.setMes( 11 );
    //Se llena el atributo año del objeto Fecha
    fechaVencimiento.setAño( 18 );
}

```

```
//Se asigna el objeto al atributo tipo Fecha de la tarjeta creada
tarjeta.setFechaVencimiento( fechaVencimiento );
return tarjeta;
}
}
```

OperacionesTarjetaCDImpl

```
package uam.edoo.laboratorioUno.operaciones;

import uam.edoo.laboratorioUno.clases.Tarjeta;
import uam.edoo.laboratorioUno.interfaces.OperacionesTarjeta;

/**
 * @author Equipo de trabajo
 *
 */
public class OperacionesTarjetaCDImpl implements OperacionesTarjeta
{

    /**
     * Se recibe la tarjeta a la que se le desea abonar el saldo
     * y el saldo que se le desea abonar, se sobre escribe
     * con el método set el saldo nuevo sumando el saldo
     * enviado y el saldo que se tiene obtenido con el
     * método ger
     */
    @Override
    public Tarjeta abonarSaldo( Tarjeta tarjeta, float saldo )
    {
        System.out.println("Abonando saldo a una tarjeta de CD");
        tarjeta.setSaldo( tarjeta.getSaldo() + saldo );
        return tarjeta;
    }

    @Override
    public void imprimirDatos( Tarjeta tarjeta )
    {
        System.out.println(tarjeta.toString());
    }

}
```

OperacionesTarjetaVideoJuegosImpl

```
package uam.edoo.laboratorioUno.operaciones;

import uam.edoo.laboratorioUno.clases.Tarjeta;
import uam.edoo.laboratorioUno.interfaces.OperacionesTarjeta;

/**
 * @author Equipo de trabajo
 *
 */
public class OperacionesTarjetaVideoJuegosImpl implements
OperacionesTarjeta
{

    @Override
    public Tarjeta abonarSaldo( Tarjeta tarjeta, float saldo )
    {
        System.out.println("Abonando saldo a una tarjeta de Video juegos");
        tarjeta.setSaldo( tarjeta.getSaldo() + saldo );
        return tarjeta;
    }

    @Override
    public void imprimirDatos( Tarjeta tarjeta )
    {
        System.out.println(tarjeta.toString());
    }

}
```

Principal

```
package uam.edoo.laboratorioUno.principal;

import java.io.*;

import uam.edoo.laboratorioUno.clases.*;
import uam.edoo.laboratorioUno.interfaces.OperacionesTarjeta;
import uam.edoo.laboratorioUno.operaciones.*;

/**
 * @author Equipo de trabajo
 *
 */
public class Principal
{

    /**
     * @param args
     */
    public static void main( String[] args )
```

```

{

//Se crea un objeto tipo Tarjeta que se llenará a partir del método
//llenarDatos(), como se llena en otro método no es necesario
//instanciarlo

LlenarDatos llenarDatos = new LlenarDatos();

//Se declaran e instancian las interfaces e implementaciones
OperacionesTarjeta operacionesCD = new OperacionesTarjetaCDImpl();
OperacionesTarjeta operacionesVideoJuegos = new
OperacionesTarjetaVideoJuegosImpl();

Tarjeta tarjeta;
TarjetaCD tarjetaCD;
TarjetaVideoJuegos tarjetaVideoJuegos;

//Se llenan los datos
tarjetaCD = llenarDatos.llenarDatosCD( "Pop" );
tarjetaVideoJuegos = llenarDatos.llenarDatosVideoJuegos();

//Se mandan a llamar los métodos correspondientes de la interfaz
operacionesCD.imprimirDatos( tarjetaCD );
operacionesVideoJuegos.imprimirDatos( tarjetaVideoJuegos );

System.out.println("Abonando saldo a las tarjetas");

/**
 * Cómo el método regresa una Tarjeta, se debe hacer un proceso
llamado
 * cast, el cuál consiste en "entender" una clase general como una
clase
 * particular
 */
tarjetaCD = ( TarjetaCD )operacionesCD.abonarSaldo( tarjetaCD, 2000F
);
tarjetaVideoJuegos = ( TarjetaVideoJuegos
)operacionesVideoJuegos.abonarSaldo( tarjetaVideoJuegos, 2000F );

operacionesCD.imprimirDatos( tarjetaCD );
operacionesVideoJuegos.imprimirDatos( tarjetaVideoJuegos );

/**
 * Se imprime el saldo
 */
System.out.println("-----");
System.out.println("Saldo CD: " + tarjetaCD.getSaldo());
System.out.println("Saldo video juego: " +
tarjetaVideoJuegos.getSaldo());

//Se instancia un objeto del tipo LlenarDatos

//Se manda a llamar el método que llena la tarjeta que se
//encuentra en la clase LlenarDatos
tarjeta = llenarDatos.llenarDatos();

```

```
//tarjetaCD = llenarDatos.llenarDatos();  
// System.out.println(tarjeta.toString());  
  
//Se manda a llamar el método que llena la tarjeta  
//desde teclado que se encuentra en la clase LlenarDatos  
//tarjeta = llenarDatos.desdeTeclado();  
  
//Se imprimen los datos usando con ayuda del método toString  
//System.out.println(tarjeta.toString());  
  
}  
  
}
```