**Ejercicio Básico POO Java. Clase Cuenta**

**EJERCICIO BÁSICO DE PROGRAMACIÓN ORIENTADA A OBJETOS EN JAVA**

Escribe una clase Cuenta para representar una cuenta bancaria. Los datos de la cuenta son: nombre del cliente (String), número de cuenta (String), tipo de interés (double) y saldo (double).

La clase contendrá los siguientes métodos:

Constructor por defecto

Constructor con todos los parámetros

Constructor copia.

Métodos setters/getters para asignar y obtener los datos de la cuenta.

Métodos *ingreso* y *reintegro*. Un ingreso consiste en aumentar el saldo en la cantidad que se indique. Esa cantidad no puede ser negativa. Un reintegro consiste en disminuir el saldo en una cantidad, pero antes se debe comprobar que hay saldo suficiente. La cantidad no puede ser negativa. Los métodos ingreso y reintegro devuelven true si la operación se ha podido realizar o false en caso contrario.

Método *transferencia* que permita pasar dinero de una cuenta a otra siempre que en la cuenta de origen haya dinero suficiente para poder hacerla. Ejemplo de uso del método transferencia:

cuentaOrigen.transferencia(cuentaDestino, importe);

que indica que queremos hacer una transferencia desde cuentaOrigen a cuentaDestino del importe indicado.

Prueba el funcionamiento de la clase Cuenta con este main:

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

String nombre, numero;

double tipo, importe;

//se crea objeto cuenta1 sin parámetros

//se ejecuta el constructor por defecto

Cuenta cuenta1 = new Cuenta();

System.out.print("Nombre : ");

nombre = sc.nextLine();

System.out.print("Número de cuenta : ");

numero = sc.nextLine();

System.out.print("Tipo de interes : ");

tipo = sc.nextDouble();

System.out.print("Saldo: ");

importe = sc.nextDouble();

cuenta1.setNombre(nombre);

cuenta1.setNumeroCuenta(numero);

cuenta1.setTipoInteres(tipo);

cuenta1.setSaldo(importe);

//se crea el objeto cuenta2 con los valores leidos por teclado

//se ejecuta el constructor con parámetros

Cuenta cuenta2 = new Cuenta("Juan Ferrández Rubio", "12345678901234567890", 1.75, 300);

//se crea cuenta3 como copia de cuenta1

//se ejecuta el constructor copia

Cuenta cuenta3 = new Cuenta(cuenta1);

//mostrar los datos de cuenta1

System.out.println("Datos de la cuenta 1");

System.out.println("Nombre del titular: " + cuenta1.getNombre());

System.out.println("Número de cuenta: " + cuenta1.getNumeroCuenta());

System.out.println("Tipo de interés: " + cuenta1.getTipoInteres());

System.out.println("Saldo: " + cuenta1.getSaldo());

System.out.println();

//se realiza un ingreso en cuenta1

cuenta1.ingreso(4000);

//mostrar el saldo de cuenta1 después del ingreso

System.out.println("Saldo: " + cuenta1.getSaldo());

//mostrar los datos de cuenta2

System.out.println("Datos de la cuenta 2");

System.out.println("Nombre del titular: " + cuenta2.getNombre());

System.out.println("Número de cuenta: " + cuenta2.getNumeroCuenta());

System.out.println("Tipo de interés: " + cuenta2.getTipoInteres());

System.out.println("Saldo: " + cuenta2.getSaldo());

System.out.println();

//mostrar los datos de cuenta3

System.out.println("Datos de la cuenta 3");

System.out.println("Nombre del titular: " + cuenta3.getNombre());

System.out.println("Número de cuenta: " + cuenta3.getNumeroCuenta());

System.out.println("Tipo de interés: " + cuenta3.getTipoInteres());

System.out.println("Saldo: " + cuenta3.getSaldo());

System.out.println();

//realizar una transferencia de 10€ desde cuenta3 a cuenta2

cuenta3.transferencia(cuenta2, 10);

//mostrar el saldo de cuenta2

System.out.println("Saldo de la cuenta 2");

System.out.println("Saldo: " + cuenta2.getSaldo());

System.out.println();

//mostrar el saldo de cuenta3

System.out.println("Saldo de la cuenta 3");

System.out.println("Saldo: " + cuenta3.getSaldo());

System.out.println();

}

**Ejercicios Básicos de Programación Orientada a Objetos Java. Clase Libro**

**EJERCICIO POO JAVA**

Crea una clase llamada Libro que guarde la información de cada uno de los libros de una biblioteca. La clase debe guardar el título del libro, autor, número de ejemplares del libro y número de ejemplares prestados. La clase contendrá los siguientes métodos:

Constructor por defecto.

Constructor con parámetros.

Métodos Setters/getters

Método *préstamo* que incremente el atributo correspondiente cada vez que se realice un préstamo del libro. No se podrán prestar libros de los que no queden ejemplares disponibles para prestar. Devuelve true si se ha podido realizar la operación y false en caso contrario.

Método *devolución* que decremente el atributo correspondiente cuando se produzca la devolución de un libro. No se podrán devolver libros que no se hayan prestado. Devuelve true si se ha podido realizar la operación y false en caso contrario.

Método *toString* para mostrar los datos de los libros. Este método se heredada de Object y lo debemos modificar (override) para adaptarlo a la clase Libro.

\*\*Esto es algo así

//método toString sobrescrito para mostrar los datos de la clase Libro

@Override

public String toString() {

return "titulo: " + titulo + "\nautor: " + autor +

"\nejemplares: " + ejemplares + "\nprestados: " + prestados;

}

Escribe un programa para probar el funcionamiento de la clase Libro.