

## Examen I Recuperación - 2ª Evaluación      21/03/2024

**Ejer 1.-** Se trata de desarrollar una aplicación Java que permita gestionar varios tipos de **cuentas bancarias**. Mediante un menú se podrán elegir determinadas operaciones:

1. **Abrir una nueva cuenta.**
2. **Ver un listado de las cuentas disponibles** (código de cuenta, titular y saldo actual).
3. **Realizar un ingreso en una cuenta.**
4. **Retirar efectivo de una cuenta.**
5. **Salir de la aplicación.**

Las cuentas se irán almacenando en alguna estructura en memoria según vayan siendo creadas. Cada cuenta será un objeto de una clase que contendrá la siguiente información:

- **Titular** de la cuenta (un objeto de la clase **Persona**, la cual contendrá información sobre el titular: **nombre, apellidos, fecha de nacimiento**).
  - **Saldo** actual de la cuenta (número real).
  - **Número de cuenta** (CCC - Código Cuenta Cliente).
  - **Tipo de interés** anual (si se trata de una **cuenta de ahorro**).
  - **Lista de entidades** autorizadas para cobrar recibos de la cuenta (si se trata de una **cuenta corriente**).
  - **Comisión de mantenimiento** (para el caso de una **cuenta corriente personal**).
  - **Tipo de interés por descubierto** (si es una **cuenta corriente de empresa**).
  - **Máximo descubierto permitido** (si se trata de una **cuenta corriente de empresa**).
- 
- Las **cuentas bancarias** pueden ser de dos tipos: **cuentas de ahorro** o bien **cuentas corrientes**.
  - Las **cuentas de ahorro** son **remuneradas** y tienen un determinado **tipo de interés**.
  - Las **cuentas corrientes** no son remuneradas.
  - Dentro de las **cuentas corrientes** podemos encontrar a su vez otros dos tipos:
    - Las **cuentas corrientes personales**, que tienen una **comisión de mantenimiento** (una cantidad fija anual).
    - Las **cuentas corrientes de empresa**, que no la tienen. Además, las **cuentas de empresa** permiten tener una cierta cantidad de **descubierto (máximo descubierto permitido)** y por tanto un **tipo de interés por descubierto** y una **comisión fija por cada descubierto** que se tenga. Es el único tipo de cuenta que permite tener **descubiertos**.

Cuando se vaya a abrir una nueva **cuenta bancaria**, el **usuario de la aplicación** (empleado del **banco**) tendrá que solicitar al **cliente**:

- **Datos personales:** **nombre, apellidos, fecha de nacimiento**(String).
- **Tipo de cuenta** que desea abrir: **cuenta de ahorro, cuenta corriente personal o cuenta corriente de empresa**.
- **Saldo inicial**.

Además de esa información, el **usuario de la aplicación** deberá también incluir:

- **Tipo de interés** de remuneración, si se trata de una **cuenta de ahorro**.
- **Comisión de mantenimiento**, si es una **cuenta corriente personal**.
- **Máximo descubierto permitido**, si se trata de una **cuenta corriente de empresa**.
- **Tipo de interés por descubierto**, en el caso de una **cuenta corriente de empresa**.
- **Comisión fija por cada descubierto**, también para el caso de una **cuenta corriente de empresa**.

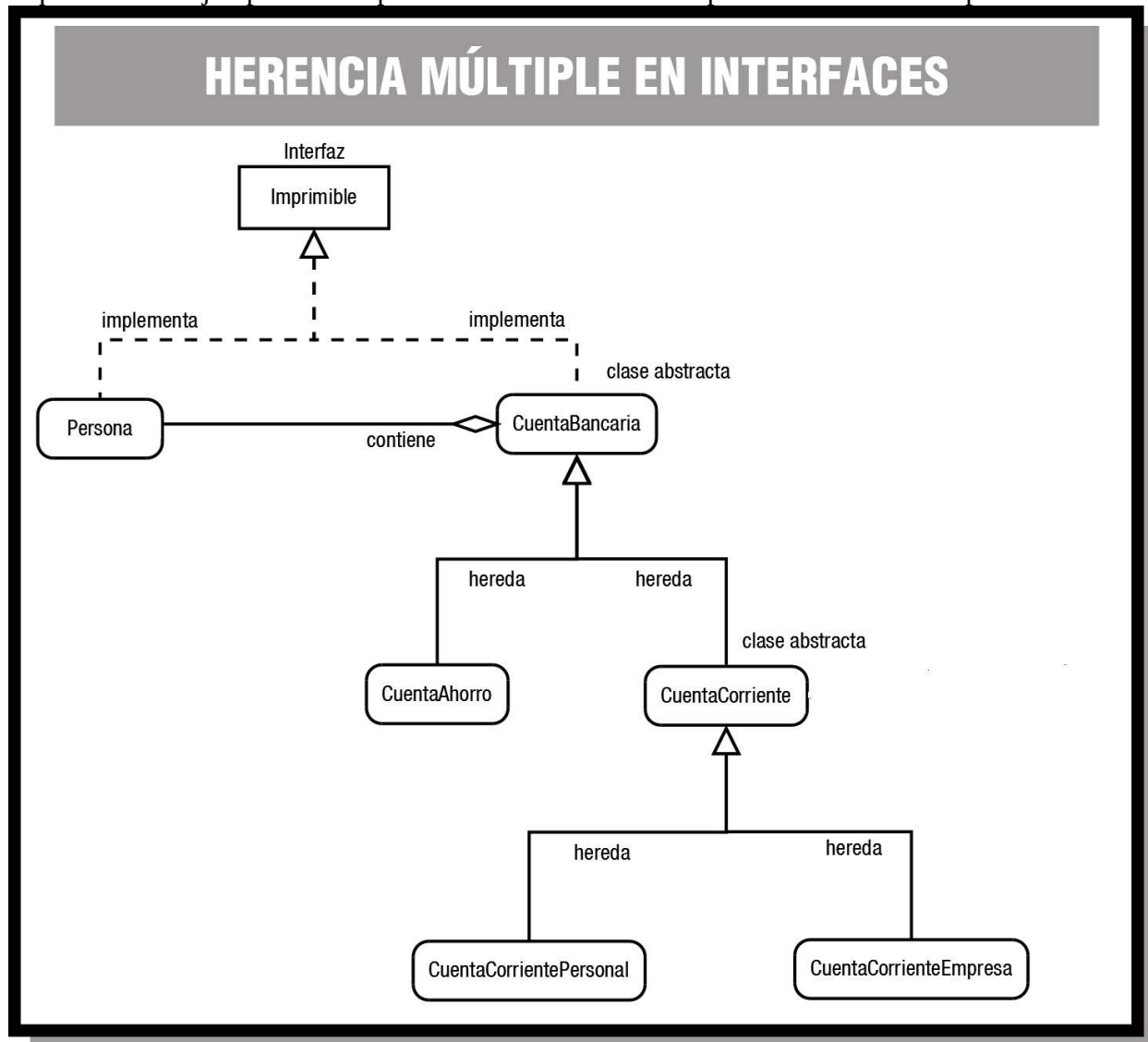
Por último el programa asignará a cada cuenta un **Número de cuenta CCC** único-

La aplicación deberá asegurarse que la información introducida sea **válida y coherente**

El programa que escribas debe cumplir al menos los siguientes requisitos:

- Para almacenar los objetos de tipo cuenta solo podrás utilizar arrays .
- Para trabajar con los datos personales, debes utilizar una clase **Persona** que contenga la información sobre los datos personales básicos del cliente (**nombre, apellidos, fecha de nacimiento**).

Aquí tienes un ejemplo de una posible estructura de clases para llevar a cabo la aplicación:



El proyecto deberá contener al menos los siguientes archivos fuente Java:

- **Programa principal** (clase con método `main`: **AplicacionCuentaBancaria.java**).
- Un archivo por cada clase o interfaz que hayas implementado.

**Ejer 2.-** Utilizando el array bidimensional `notasAlumnos[][]`, escribe un programa que muestre por pantalla las medias de los alumnos para cada una de sus asignaturas.

```
notasAlumnos = {{"Luis","Programacion","6.75","4","5.8"},  
                {"Luis","Sistemas","2.75","6","5.9"},  
                {"Luis","Lenguajes de marcas","9","7","6.1","3.85"},
```

```
{ "Laura", "Programacion", "2.75", "3", "8"},
{ "Laura", "Sistemas", "2.75", "6", "3.9"},
{ "Laura", "Lenguajes de marcas", "1", "4.25", "5", "6.5"},
{ "Alba", "Programacion", "6", "4.5", "9.8"},
{ "Alba", "Sistemas", "8.75", "9", "7.9"},
{ "Alba", "Lenguajes de marcas", "9", "10", "6.1", "7" } };
```

Ejemplo de salida:

*Alumno: Luis Materia: Programacion Media: 5.516666666666667*

*Alumno: Luis Materia: Sistemas Media: 4.883333333333334*

*Alumno: Luis Materia: Lenguajes de marcas Media: 6.487500000000001*

*Alumno: Laura Materia: Programacion Media: 4.583333333333333*

*Alumno: Laura Materia: Sistemas Media: 4.216666666666667*

*Alumno: Laura Materia: Lenguajes de marcas Media: 4.1875*

*Alumno: Alba Materia: Programacion Media: 6.766666666666667*

*Alumno: Alba Materia: Sistemas Media: 8.549999999999999*

*Alumno: Alba Materia: Lenguajes de marcas Media: 8.025*

### **Criterios de puntuación. Total 10 puntos.**

Para poder empezar a aplicar estos criterios es necesario que la aplicación compile y se ejecute correctamente en un ordenador. En caso contrario la puntuación será directamente de 0,00.

Criterios de puntuación.

Ejercicio 1	8,00
Ejercicio 2	2,00
<b>Total (máximo)</b>	<b>10,00</b>

### **Entrega.**

Una vez que tengas terminados el programa (**carpeta de proyecto Netbeans**, incluyendo todos los archivos fuente y todos los recursos) comprímelo en **un único archivo**. El envío se realizará **a través de la plataforma** de la forma establecida para ello, y el archivo se nombrará siguiendo las siguientes pautas:

**apellido1\_apellido2\_nombre\_SIG\_Examen\_2\_I\_Rec**

Así por ejemplo la alumna **Begoña Sánchez Mañas**, debería nombrar este archivo como...

**sanchez\_manas\_begona\_PROG\_Examen\_2\_I\_Rec**