

EJERCICIO REPASO CLASES Y HERENCIA

Definir la clase base **Cuenta**, que contiene las siguientes propiedades:

- Numero (la identifica y se genera de forma automática).
- Titular string
- Dni string
- FechaNac date
- Saldo double. (No se podrá modificar directamente, sólo ingresando o retirando dinero)

• Crear **2 constructores**:

- uno que por defecto que inicialice los atributos de cada clase.
- otro deberá permitir ingresar todos los atributos para cada clase. Hay que validar las entradas de datos.

• Usar **Getters y Setters** para cada atributo de las clases a crear.

• Crear el método **Reintegro** que permita retirar dinero de la cuenta, este método recibirá como parámetro el importe que desea retirar. Se debe comprobar que hay saldo suficiente. No se permiten reintegros superiores al saldo.

• Crear el método **Ingreso** que permita hacer un deposito en una cuenta bancaria, este método recibirá como parámetro el importe que se desea depositar. Esa cantidad no puede ser negativa.

• Los métodos Ingreso y Reintegro devuelven true si la operación se ha podido realizar o false en caso contrario.

• Crear un método **ToString** que permita imprimir todos los atributos de la clase.

• Crear método **Transferencia** que permita pasar dinero de una cuenta a otra siempre que en la cuenta de origen haya dinero suficiente para poder hacerla, recibiendo como argumentos la cuentaDestino e Importe.

• Crear un método **esTitularValido()** que a partir de FechaNac devuelve verdadero si el titular es mayor de edad y menor de 25 años, (en este caso pasará a crearse CuentaJoven de forma automática) y falso en caso contrario.

Definir también la clase **CuentaJoven** que deriva de Cuenta.

Tendrá una propiedad adicional que será:

- Bonificación(%) long
- Además el reintegro de dinero sólo se podrá realizar si el titular es válido.
- Crear método **IngresoBonifica** que compruebe el importe, si este es superior a 1000€ se incrementará también el saldo con el % de la bonificación sobre el importe a ingresar.
- El método **Mostrar** mostrará el mensaje de "Cuenta Joven", el % de la bonificación y el valor del saldo.

Definir la clase **CuentaEmpresa** que deriva de Cuenta.

Posee una línea de crédito, de la cual, si se desea retirar dinero de la cuenta y la cantidad a retirar es mayor al saldo de la cuenta, esta puede hacer uso de su línea de crédito.

- Interés long
- Crédito double (No se podrá modificar directamente, sólo al retirar dinero)
- mesesCredito int
- Solo se podrá conceder crédito cuando se trate de una CuentaEmpresa.
- Cuando en un Reintegro el saldo sea inferior al importe a retirar, se invocará al método **CalcularCredito** que concederá crédito por la diferencia y esta diferencia se incrementará al valor de la propiedad Crédito.

- Crear método **MostrarCredito** que recibirá como parámetro **mesesCredito** y mostrará el importe del credito, el interes y la cuota mensual $(=(\text{credito} + (\text{credito} * \text{interes})) / \text{mesesCredito})$

Definir un objeto List llamado **LasCuentas**.

Crear **menú** para poder realizar las siguientes operaciones sobre la List.

- Alta de una cuenta, se guardará en cuentasList.
- Consulta de una cuenta, a través del nº de cuenta o Dni
- Operaciones Ingresos sobre las cuentas.
- Operaciones Reintegros sobre las cuentas.
- Transferencias entre cuentas.
- Salir de la aplicación.

`objeto.GetType ();` → Obtiene el **Type** de la instancia actual.