

Actividades – herencia

Primer ejercicio

Una universidad quiere registrar la información de sus alumnos y docentes. Para eso pensaron en usar una aplicación que usa los atributos de la programación orientada a objetos. Se pide realizar las siguientes clases:

- Persona: Debe tener los atributos legajo, nombre, apellido y e-mail. Los métodos getters y setters para esos atributos y un método mostrarDatos(). La clase persona debe implementar la interfaz IEquatable para poder buscar personas, alumnos o profesores según su legajo.
- Alumno: Debe tener además de los atributos y métodos de la clase persona los atributos cantidad de materias aprobadas, beca (porcentaje) y un atributo de clase que represente el valor de la cuota del alumno. Debe sobrescribir el método de la clase persona y además debe tener un método calcularCuota que descuenta del valor de la cuota el porcentaje de beca del alumno. Debe implementar la interfaz IComparable para poder comparar alumnos según su cuota.
- Profesor: Debe tener además de los atributos de la clase persona los atributos SalarioBasico (atributo de clase) y antigüedad. Debe sobrescribir el método mostrarDatos de la clase persona. Además debe tener un método calcularSalario que le sume al salario básico un 3% por año de antigüedad del docente. Debe implementar la interfaz IComparable para poder comparar profesores según su salario.

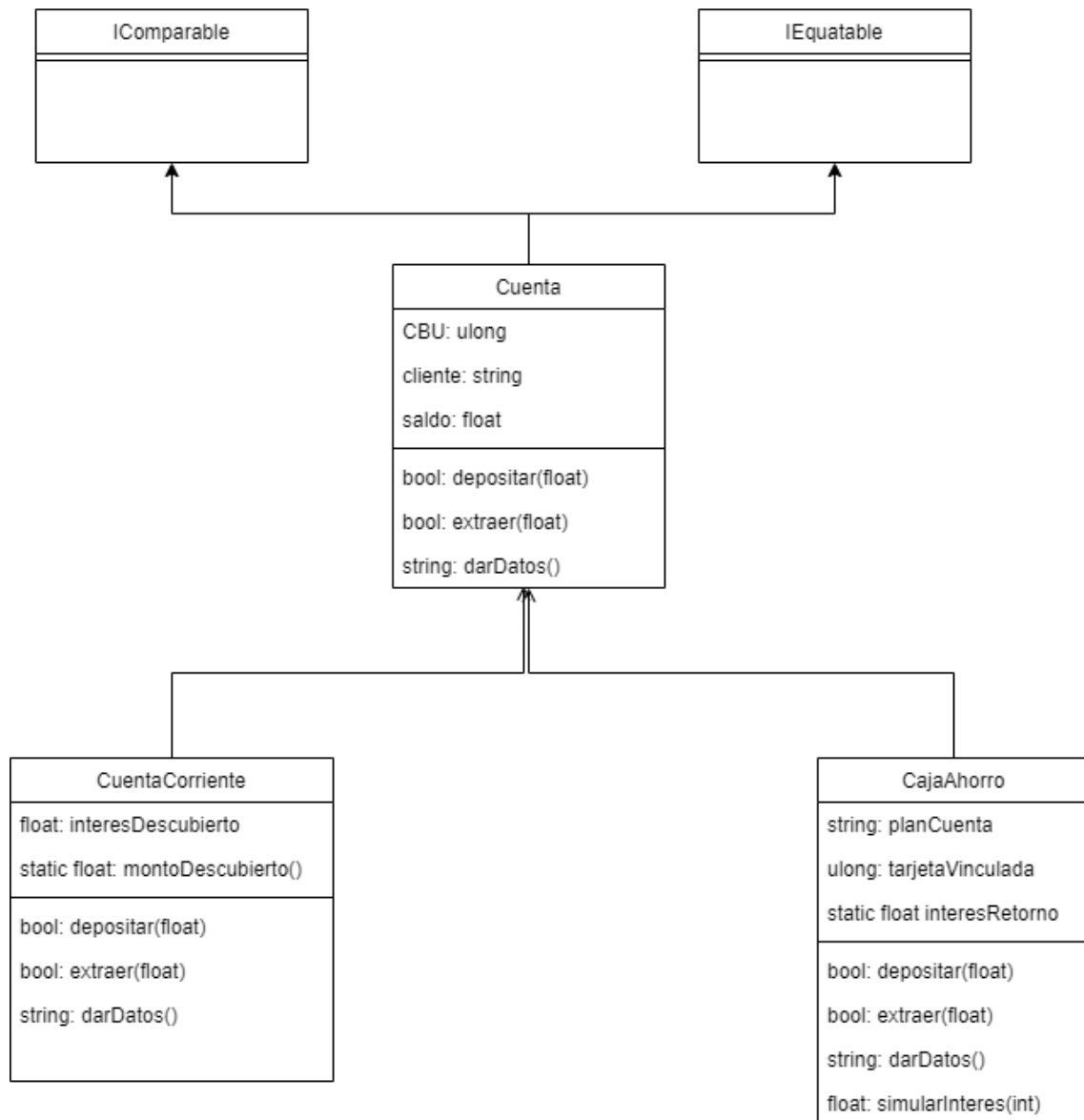
Se deben pedir al inicio del programa el ingreso para setear los atributos de clase para poder trabajar. Una vez ingresados los mismos se debe pedir el ingreso primero de alumnos y luego de profesores en dos listas separadas. Una vez terminado el ingreso se deberán mostrar el listado ordenado de alumnos y profesores.

Segundo ejercicio

Se tienen dos tipos de cuentas bancarias en un banco:

- Las cajas de ahorro: tienen vinculados un plan de cliente y una tarjeta de débito aparte de los datos que viene teniendo una cuenta bancaria. No puede realizar extracciones en descubierto y además tiene intereses por tener el dinero depositado en la caja de ahorro.
- Las cuentas corrientes: No tienen vinculados a las mismas ni tarjetas ni planes de cliente. Aparte, pueden tener extracciones en descubierto siempre y cuando el monto sea menor al máximo monto en descubierto que se pueda extraer. O sea, el monto puede quedar en negativo después de una extracción siempre y cuando sea menor al monto en descubierto (si el saldo me quedaría en \$-1500 y el monto en descubierto es \$1000, la extracción no se puede hacer). Además, al realizar un nuevo depósito, si el monto es negativo se debe restar al monto a depositar un interés en descubierto (por ejemplo, si el monto a depositar es 1000, el interés en descubierto es 10% y el saldo actual es menor a 0, se deberá depositar \$1000 menos el 10%).

El diagrama de clases final será:



Crear un programa que permita crear cuentas bancarias (ya sean cuentas corrientes o cajas de ahorro) y agregarlas a una lista de cuentas. El ingreso de cuentas debe finalizar con el CBU 0.

Una vez finalizado el ingreso de datos se debe permitir ingresar el CBU y debe permitir mostrar los datos, realizar una extracción y un depósito. El ingreso de estas cuentas también debe finalizar con el CBU 0.

Tercer ejercicio

Modificar el programa de la aplicación bancaria para poder ingresar Cajas de ahorro y cuentas corrientes.

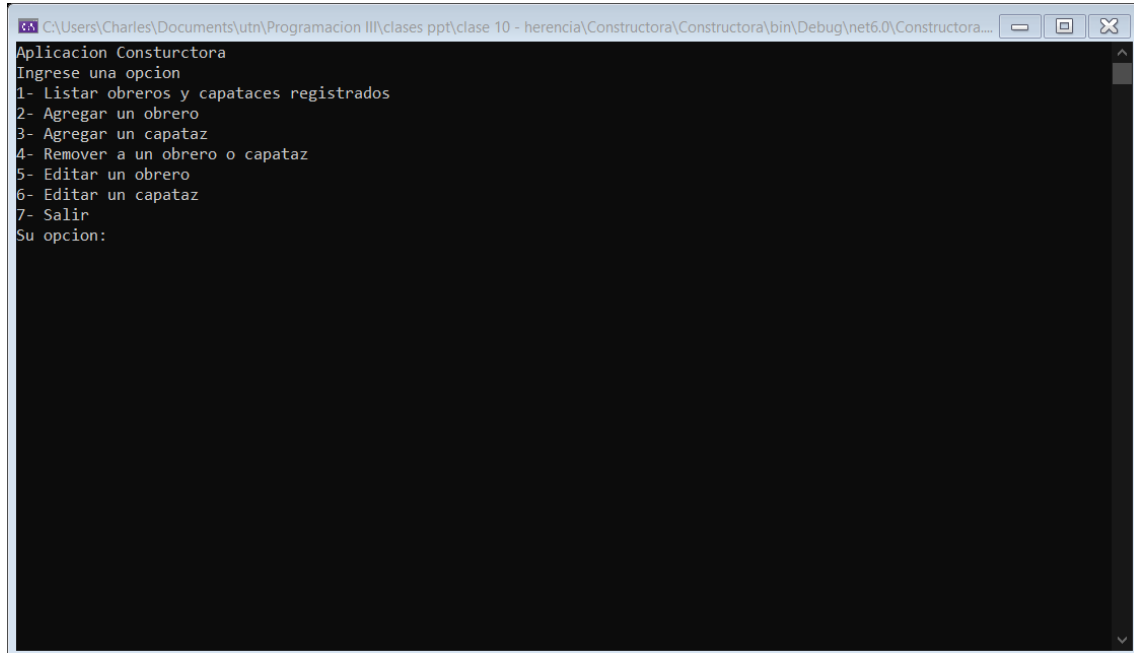
Cuarto ejercicio

Una constructora tiene distintos empleados que pueden ser obreros o capataces. Se pide armar un sistema de gestión que permita guardar en una lista a los obreros y capataces de la obra. Se deben poder realizar las siguientes acciones:

- Registrar obreros en la constructora, con un DNI (cadena de caracteres), un apellido, un nombre y una especialidad ("Albañil", "Pintor", "Plomero", "Herrero" o "Electricista").

- Registrar capataces en la constructora, indicando para los mismos, un DNI (cadena de caracteres), un apellido, un nombre y un número de matrícula profesional (entero sin signo).
- Remover a un obrero o capataz de la constructora.
- Editar datos de un capataz buscando por DNI e ingresando los nuevos datos
- Editar datos de un obrero buscando por DNI e ingresando los nuevos datos.

La interfaz del programa debe mostrar un menú como el siguiente:



```

C:\Users\Charles\Documents\utn\Programacion III\clases ppt\clase 10 - herencia\Constructora\Constructora\bin\Debug\net6.0\Constructora...
Aplicacion Consturctora
Ingrese una opcion
1- Listar obreros y capataces registrados
2- Agregar un obrero
3- Agregar un capataz
4- Remover a un obrero o capataz
5- Editar un obrero
6- Editar un capataz
7- Salir
Su opcion:
  
```

Se deberá implementar:

- Herencia (tener presente que tanto obreros como capataces son empleados).
- Interfaces (usar IEquatable para poder buscar).
- List o ArrayList (según prefiera el alumno).
- Getters y setters
- Constructores por defecto y parametrizados.
- Clases controladoras y de interfaces (opcional).