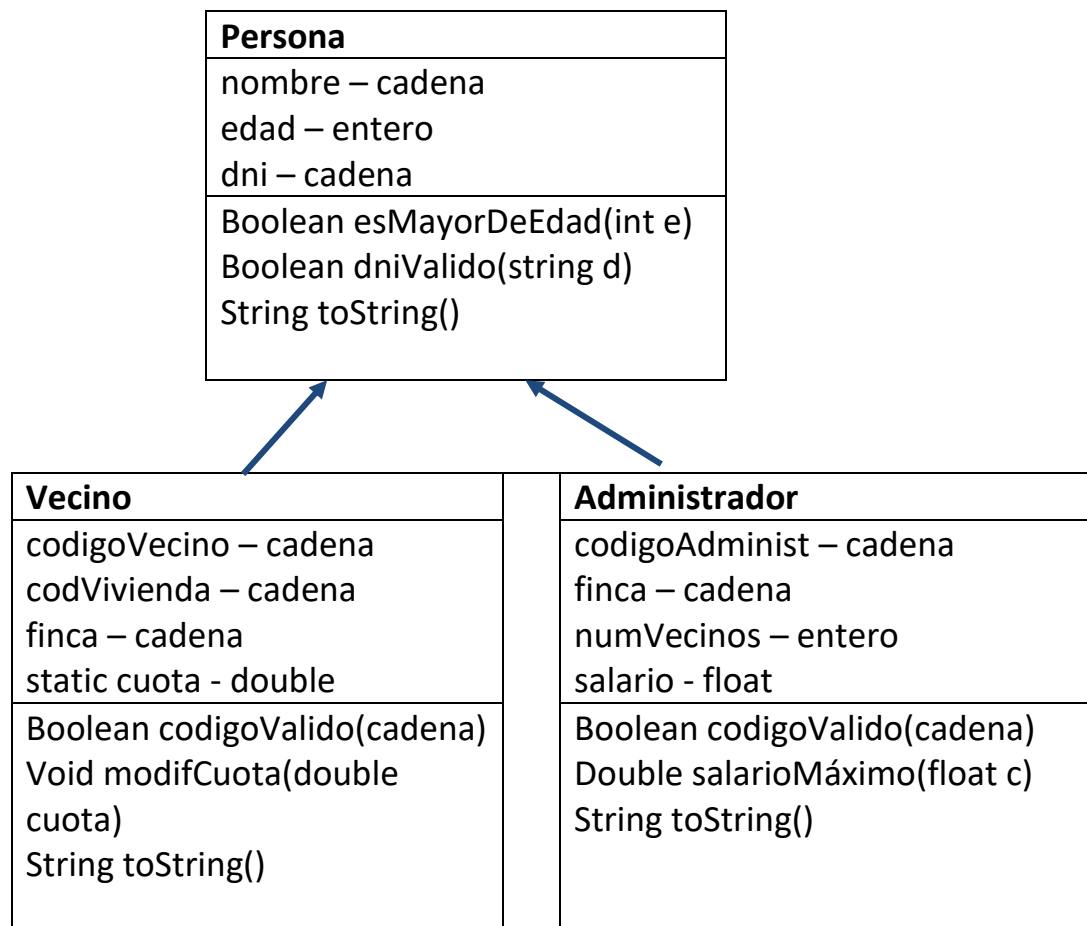


PRÁCTICA I I_3

HERENCIA III

Programa la siguiente jerarquía de clases:



**(*) Todos los atributos son private.
Programa get y set necesarios.**

Constructores:

- Por defecto
- Otro con los valores de todos los atributos
- En el caso de la clase Vecino, además un constructor que no contenga el DNI.

Métodos/funciones:

- **esMayorDeEdad** – comprueba si el parámetro edad es mayor o igual que 18. Devuelve true o false.
- **dniValido** – Comprueba si el atributo DNI tiene un contenido válido, devuelve true si es así o false si no. Todos los DNI deben tener los 8 dígitos numéricos y las letras que correspondan. Te adjunto un enlace donde se explica cómo se calcula el dígito de control.
<http://www.interior.gob.es/web/servicios-al-ciudadano/dni/calculo-del-digito-de-control-del-nif-nie>
- **codigoValido** en Vecino – máximo 5 caracteres y debe comenzar por la letra V.
- **codigoValido** en Administrador – máximo 5 caracteres y debe comenzar por la letra A.
- **actualizarCuota** : método que recibe un valor double que utiliza para modificar el valor del atributo cuota que es static, es decir, el nuevo valor tendrá validez para todos los objetos de la clase vecino. No devuelve nada.
- **salarioMáximo** debe calcular el importe máximo del salario del administrador que será igual al 50% del numVecinos*cuota (c).
- **toString()** en las 3 clases.

Programa un método main() que haga lo siguiente:

- Instancia un objeto Vecino con edad 16 años y el resto de los valores inventados. (utiliza el constructor que no incorpora el DNI).
A continuación pide el valor de DNI y programa un bucle que se repita mientras en dni introducido no sea un dni válido. Utiliza el método dniValido(). Cuando salgamos del bucle con un DNI válido le asignaremos al atributo dni del objeto creado anteriormente el valor introducido por teclado.
- Comprueba si el vecino es mayor de edad y si no lo es sustituye edad por 18.
- Instancia un objeto Administrador con edad 16 años y el resto de los valores inventados. (utiliza los constructores)
- Comprueba si es mayor de edad y si no lo es sustituye edad por 18.
- Comprueba si el salario que hemos asignado al objeto es mayor que el salario máximo que nos calcula el método salarioMáximo(). A este método le pasaremos como parámetro un valor para cuota. Esta cuota la obtendremos del objeto de la clase Vecino, es uno de sus atributos. Si

salario es mayor que salarioMáximo sustituye el atributo salario del objeto Administrador por el valor devuelto por la función.

- Utiliza el método toString() de las 3 clases, cómo lo harías. Observa la diferencia de los resultados.