

BOLETIN5_Ejercicios Básicos de Programación Orientada a Objetos

1) Crea una clase **Contador** con un atributo entero "contador" y con los métodos para **incrementar** y **decrementar** el contador. La clase contendrá un constructor por defecto, un constructor con parámetros y los métodos getters y setters.

2) Crea una clase llamada **Cuenta** que tendrá los siguientes atributos: titular y cantidad (puede tener decimales).

- El titular será obligatorio y la cantidad es opcional. Crea dos constructores que cumpla lo anterior
- Crea sus métodos get, set y toString.
- Tendrá dos métodos especiales:
 - **ingresar(double cantidad): void.** se ingresa una cantidad a la cuenta, si la cantidad introducida es negativa, no se hará nada.
 - **retirar(double cantidad): void.** se retira una cantidad a la cuenta, si restando la cantidad actual a la que nos pasan es negativa, la cantidad de la cuenta pasa a ser 0.

3) Crea una clase **Fecha**. La clase contendrá

- Además de constructores, métodos set y get y el método **toString**.
- Método para comprobar si la fecha es correcta.
- Método para modificar la fecha actual por la del día siguiente.

4) Escribir una clase que represente un **reloj** que señale la hora el minuto, y el segundo. La clase dispondrá de dos constructores, uno sin parámetros, que pone el reloj a 0:0:0, y otro al que se le pasa la hora, los minutos y los segundos. Además, habrá que realizar los siguientes métodos:

- **dameHora(): String.** Uno que da la hora, los minutos y los segundos, separados por el carácter ":", en una cadena.
- **dameHora(String tipo).** Dos tipos: "24hrs" y "12hrs". Otro que también da la hora, pero en formato 24 horas (como el anterior) o en formato de 12 horas, en cuyo caso debe distinguir entre "am" (para horas de 0 a 11) o "pm", para horas de 12 a 23, también en una cadena.

5) Haz una clase llamada **Persona** que siga las siguientes condiciones:

- Sus atributos son: nombre, edad, DNI, sexo (H hombre, M mujer), peso y altura. No queremos que se accedan directamente a ellos.
- Por defecto, todos los atributos menos el DNI serán valores por defecto según su tipo (0 números, cadena vacía para String, etc.). Sexo sera hombre por defecto, usa una constante para ello.

Se implantarán varios constructores:

- Un constructor por defecto.
- Un constructor con el nombre, edad y sexo, el resto por defecto.
- Un constructor con todos los atributos como parámetro.

Los métodos que se implementaran son:

- `calcularIMC()`: calcula si la persona esta en su peso ideal (peso en kg/(altura² en m)), si esta fórmula devuelve un valor menor que 20, la función devuelve un -1, si devuelve un número entre 20 y 25 (incluidos), significa que esta por debajo de su peso ideal la función devuelve un 0 y si devuelve un valor mayor que 25 significa que tiene sobrepeso, la función devuelve un 1. Te recomiendo que uses constantes para devolver estos valores.
- `esMayorDeEdad()`: indica si es mayor de edad, devuelve un booleano.
- `comprobarSexo(char sexo)`: comprueba que el sexo introducido es correcto. Si no es correcto, sera H. No sera visible al exterior.
- `toString()`: devuelve toda la información del objeto.
- `generaDNI()`: genera un número aleatorio de 8 cifras, genera a partir de este su número su letra correspondiente. Este método sera invocado cuando se construya el objeto. Puedes dividir el método para que te sea más fácil. No será visible al exterior.

Métodos set de cada parámetro, excepto de DNI.

Ahora, crea una clase ejecutable que haga lo siguiente:

- Pide por teclado el nombre, la edad, sexo, peso y altura.
- Crea 3 objetos de la clase anterior, el primer objeto obtendrá las anteriores variables pedidas por teclado, el segundo objeto obtendrá todos los anteriores menos el peso y la altura y el último por defecto, para este último utiliza los métodos set para darle a los atributos un valor.
- Para cada objeto, deberá comprobar si está en su peso ideal, tiene sobrepeso o por debajo de su peso ideal con un mensaje.
- Indicar para cada objeto si es mayor de edad.
- Por último, mostrar la información de cada objeto.