

## 1- **Nos piden hacer que gestionemos una serie de productos.**

Los productos tienen los siguientes atributos:

- Nombre
- Precio

Tenemos dos tipos de productos:

- Perecedero: tiene un atributo llamado días a caducar
- No perecedero: tiene un atributo llamado tipo

Crea sus constructores, getters, setters y toString.

Tendremos una función llamada calcular, que según cada clase hará una cosa u otra, a esta función le pasaremos un numero siendo la cantidad de productos

- En Producto, simplemente seria multiplicar el precio por la cantidad de productos pasados.
- En Perecedero, aparte de lo que hace producto, el precio se reducirá según los días a caducar:
  - Si le queda 1 día para caducar, se reducirá 4 veces el precio final.
  - Si le quedan 2 días para caducar, se reducirá 3 veces el precio final.
  - Si le quedan 3 días para caducar, se reducirá a la mitad de su precio final.
- En NoPerecedero, hace lo mismo que en producto

Crea una clase ejecutable y crea un array de productos y muestra el precio total de vender 5 productos de cada uno. Crea tú mismo los elementos del array.

## **2- Se pide hacer un programa que gestione empleados.**

Los empleados se definen por tener:

- Nombre
- Edad
- Salario

También tendremos una constante llamada PLUS, que tendrá un valor de 300€

Tenemos dos tipos de empleados: repartidor y comercial.

El comercial, aparte de los atributos anteriores, tiene uno más llamado comisión (double).

El repartidor, aparte de los atributos de empleado, tiene otro llamado zona (String).

Crea sus constructores, getters and setters y toString (piensa como aprovechar la herencia).

No se podrán crear objetos del tipo Empleado (la clase padre) pero si de sus hijas.

Las clases tendrán un método llamado plus, que según en cada clase tendrá una implementación distinta. Este plus básicamente aumenta el salario del empleado.

- En comercial, si tiene más de 30 años y cobra una comisión de más de 200 euros, se le aplicara el plus.
- En repartidor, si tiene menos de 25 y reparte en la “zona 3”, este recibirá el plus.

Puedes hacer que devuelva un booleano o que no devuelva nada, lo dejo a tu elección.

Crea una clase ejecutable donde crees distintos empleados y le apliques el plus para comprobar que funciona.

### **3- Nos piden realizar una agenda telefónica de contactos.**

Un contacto está definido por un nombre y un teléfono (No es necesario de validar). Un contacto es igual a otro cuando sus nombres son iguales.

Una agenda de contactos está formada por un conjunto de contactos (Piensa en que tipo puede ser)

Se podrá crear de dos formas, indicándoles nosotros el tamaño o con un tamaño por defecto (10)

Los métodos de la agenda serán los siguientes:

- `añadirContacto(Contacto c)`: Añade un contacto a la agenda, sino se pueden meter más a la agenda se indicara por pantalla. No se pueden meter contactos que existan, es decir, no podemos duplicar nombres, aunque tengan distinto teléfono.
- `existeContacto(Contacto c)`: indica si el contacto pasado existe o no.
- `listarContactos()`: Lista toda la agenda
- `buscaContacto(String nombre)`: busca un contacto por su nombre y muestra su teléfono.
- `eliminarContacto(Contacto c)`: elimina el contacto de la agenda, indica si se ha eliminado o no por pantalla
- `agendaLlena()`: indica si la agenda está llena.
- `huecosLibres()`: indica cuantos contactos más podemos meter.

Crea un menú con opciones por consola para probar todas las funcionalidades.

4- Crearemos una supeclase llamada Electrodomestico con las siguientes características:

- Sus atributos son **precio base**, **color**, **consumo energético** (letras entre A y F) y **peso**. Indica que se podrán heredar.
- Por defecto, el color será blanco, el consumo energético será F, el precioBase es de 100 € y el peso de 5 kg. Usa constantes para ello.
- Los colores disponibles son blanco, negro, rojo, azul y gris. No importa si el nombre está en mayúsculas o en minúsculas.
- Los constructores que se implementarán serán:
  - Un constructor por defecto. Un constructor con el precio y peso. El resto por defecto. Un constructor con todos los atributos.
- Los métodos que implementara serán:
  - Métodos get de todos los atributos.
  - **comprobarConsumoEnergetico(char letra)**: comprueba que la letra es correcta, sino es correcta usara la letra por defecto. Se invocará al crear el objeto y no sera visible.
  - **comprobarColor(String color)**: comprueba que el color es correcto, sino lo es usa el color por defecto. Se invocará al crear el objeto y no sera visible.
  - **precioFinal()**: según el consumo energético, aumentara su precio, y según su tamaño, también. Esta es la lista de precios:

LETRA	PRECIO
A	100 €
B	80 €
C	60 €
D	50 €
E	30 €
F	10 €

TAMAÑO	PRECIO
Entre 0 y 19 kg	10 €
Entre 20 y 49 kg	50 €
Entre 50 y 79 kg	80 €
Mayor que 80 kg	100 €

Crearemos una subclase llamada **Lavadora** con las siguientes características:

- Su atributo es **carga**, además de los atributos heredados.
- Por defecto, la carga es de 5 kg. Usa una constante para ello.
- Los constructores que se implementarán serán:
  - Un constructor por defecto.
  - Un constructor con el precio y peso. El resto por defecto.
  - Un constructor con la carga y el resto de los atributos heredados.

Recuerda que debes llamar al constructor de la clase padre.

- Los métodos que se implementara serán:
  - Método get de carga.
  - **precioFinal()**: si tiene una carga mayor de 30 kg, aumentara el precio 50 €, sino es así no se incrementara el precio. Llama al método padre y añade el código necesario. Recuerda que las condiciones que hemos visto en la clase Electrodomestico también deben afectar al precio.

Crearemos una subclase llamada **Televisión** con las siguientes características:

- Sus atributos son **resolución** (en pulgadas) y **sintonizador TDT** (booleano), además de los atributos heredados.
- Por defecto, la resolución será de 20 pulgadas y el sintonizador será false.

- Los constructores que se implementarán serán:
  - Un constructor por defecto.
  - Un constructor con el precio y peso. El resto por defecto.
  - Un constructor con la resolución, sintonizador TDT y el resto de atributos heredados. Recuerda que debes llamar al constructor de la clase padre.
- Los métodos que se implementara serán:
  - Método get de resolución y sintonizador TDT.
  - **precioFinal()**: si tiene una resolución mayor de 40 pulgadas, se incrementara el precio un 30% y si tiene un sintonizador TDT incorporado, aumentara 50 €. Recuerda que las condiciones que hemos visto en la clase Electrodomestico también deben afectar al precio.

Ahora crea una clase ejecutable que realice lo siguiente:

- Crea un array de Electrodomesticos de 10 posiciones.
- Asigna a cada posición un objeto de las clases anteriores con los valores que desees.
- Ahora, recorre este array y ejecuta el método precioFinal().
- Deberás mostrar el precio de cada clase, es decir, el precio de todas las televisiones, por un lado, el de las lavadoras por otro y la suma de los Electrodomesticos (puedes crear objetos Electrodomestico, pero recuerda que Television y Lavadora también son electrodomésticos). Recuerda el uso operador instanceof.

Por ejemplo, si tenemos un Electrodomestico con un precio final de 300, una lavadora de 200 y una televisión de 500, el resultado final sera de 1000 (300+200+500) para electrodomésticos, 200 para lavadora y 500 para televisión.