

Unidad 3: Programación Orientada a Objetos 1

Ejercicios boletín 2

NOTA: Todas las operaciones se realizarán mediante métodos.

Arrays de objetos

1. Escriba una clase Producto, caracterizada por 5 atributos (uno de ellos debe ser precio de fábrica). En una clase Gestion, con un array de productos como atributo se podrá: listar datos de todos los productos, calcular el precio de venta al público, que será el precio de fábrica más un 50 %, comprobar si el producto es “frágil” o no, mediante un método que devuelva un tipo boolean y añadir un producto a la lista pasando como parámetro un producto. Crear un main de prueba.

2. Estoy en crisis. He pensado en alquilar las habitaciones de mi casa, pero estilo hotel para que tenga caché. Debemos gestionar mi “minihotel” y para ello, debemos crear la clase Habitación, una clase Hotel con una lista de Habitaciones y una clase de prueba para gestionar el hotel (realmente habría algunas clases más, como serían Cliente, Productos, etc. Pero no entraremos en eso). Se podrá crear una habitación (cuyos atributos serán tipo, precio, qué servicios extra tiene, nombre de cliente asignado, número de días contratados, etc.) ver si está ocupada o no, calcular precio final según número de días, servicios extra contratados y consumo del minibar, mostrar factura, comprobar si está “limpia” (que ha pasado la limpiadora cada día) y aquellos métodos que se quieran añadir. Crear un main de prueba.

3. Vamos a intentar modelar a un vendedor ambulante de “móviles”. Los atributos de la clase Móvil serán, como mínimo, marca, modelo, vendido o no y precio unitario. En la clase Vendedor con un array de Móviles y total vendido, como atributos mínimos, se deben crear métodos que hagan las siguientes operaciones: Comprobar que le quedan móviles, mostrar cuántos móviles le quedan sin vender, mostrar precio final de un producto haciendo algún descuento al precio unitario, etc. Podemos incluir también, si se quiere, la posibilidad de ofrecer cambio. Escribir también una clase de prueba para hacer todas las operaciones.

4. Crear un programa con una clase Cliente, caracterizado por dni, nombre, apellidos, activo o no, peso y altura. En esta clase, se podrá calcular el índice de masa corporal (IMC) de un Cliente (peso en $\text{kg}/\text{altura}^2$ en m). Crear la clase Gimnasio donde tendremos una lista de cliente como característica y los métodos:

Calcular IMC al cliente elegido de la lista buscando el cliente, mostrar lista, añadir cliente a la lista, dar de baja, calcular la media de IMC entre todos los clientes. Se pueden calcular otras medias como de altura, peso, solo de los mayores de una determinada edad...

Crear un main para probar todo.

5. Crearemos una clase llamada Electrodomestico con las siguientes características:

Sus atributos son precio base, color, consumo energético (letras entre A y F) y peso.

Por defecto, el color será blanco, el consumo energético sera F, el precioBase es de 100 € y el peso de 5 kg.

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 implementará serán:

Métodos get de todos los atributos.

comprobarConsumoEnergetico (char letra): comprueba que la letra es correcta, si no es correcta usará la letra por defecto. Se invocará al crear el objeto.

comprobarColor (String color): comprueba que el color es correcto, si no lo es usa el color por defecto. Se invocará al crear el objeto.

precioFinal(): según el consumo energético, aumentará su precio en la cantidad que se puede ver en la tabla. Esta es la lista de precios:

Letra	A	B	C	D	E	F
Precio (€)	100	80	60	50	30	10

Crear una clase tienda con un array de electrodomésticos como atributo y se pueda calcular el precio de un objeto elegido por el usuario, la suma de todos los precios de todos los objetos del array y la media de precio entre todos los objetos de la lista.

Crea un main para poder probar todo.

6. Este ejercicio debería ser mucho más complejo, pero solo debemos hacer lo que se pide, aunque en alguna ocasión tenga menos sentido o falten cosas por hacer. Se trata de implementar un programa para gestionar un cine con una sola sala.

Cada Entrada (una de las clases) de cine tendrá un número identificativo, precio, si está libre u ocupada, número de la fila y de butaca. Todas las funcionalidades estarán en la clase Sala, que estará caracterizada por número identificador, número total de localidades, un objeto tipo Entrada y un array de localidades (tipo Entrada).

Se darán al usuario las siguientes opciones en un switch:

- Ver cartelera que mostrará el título de la película proyectada.

- Comprar una entrada comprobando si una localidad está vendida ya anteriormente. Todas las entradas valen lo mismo. En este caso, solo hay que mirar si su atributo "libre" es verdadero o falso, sin preocuparnos por nada más. Al acabar una venta habrá que pasar esa entrada a ocupada.
- Calcular ganancias totales del día. Para ello al final de cada compra habrá que sumar la cantidad en el lugar que se haya habilitado para llevar el dinero total en caja.
- Modificar el precio de una entrada.
- Alguno más que se te ocurra.

De repaso

7. (Repaso métodos, una sola clase y el main) Defina una clase 'Monedero' que permita gestionar la cantidad de dinero de que una persona dispone en un momento dado.

La clase deberá tener un constructor que permitirá crear un monedero con una cantidad de dinero inicial y deberá definir un método para meter dinero en el monedero, otro para sacarlo y finalmente, otro para consultar el disponible; solo podrá conocerse la cantidad de dinero del monedero a través de este último método. Por supuesto, no se podrá sacar más dinero del que haya en un momento dado en el monedero.

Para probar el funcionamiento de la clase, escriba un método 'main' con una serie de instrucciones que hagan uso de los métodos definidos.

8. Puedes crear ejercicios de repaso mediante una sola clase modelo y los métodos que hagan algún cálculo sobre sus atributos, por ejemplo:

Productos y precios (rebajar por días para caducar, descuentos, etc.)

Trabajadores y sueldos (pago de impuestos, extras por vender mucho...)

Series, películas o redes sociales (medias de número de likes, más vistos, menos, listar por secciones (terror, comedia...))

Alumnos y notas (medias, suspensos, calculo de nota final mediante ponderación de exámenes, trabajos...)

Todos ellos pueden ser solo de dicha clase modelo y el main o con más clases, como por ejemplo, Alumno-Aula-main, Trabajador-Oficina-main , Productos-Tienda-main...