

## Unidad 3 Boletín ejercicios 1

### Programación Orientada a Objetos 1

**NOTA:** Todas las operaciones se realizarán mediante métodos y todos los ejercicios tendrán una clase principal donde se comprueben todas las acciones que haga el programa. Nunca leeremos dentro de los métodos.

Además, estos ejercicios se harán sin usar arrays de objetos que se verán más adelante.

1. Hacer un programa que calcule el área de un círculo en  $\text{cm}^2$ . En el main debemos crear un objeto tipo Círculo y llamar a dicho método. Mejorar el programa añadiendo a la clase Círculo otro método que calcule el área del círculo en  $\text{m}^2$ .
2. Implementar un programa que contenga dos métodos en una clase "operaciones": un método al que se le pase un número entero y diga si este es positivo o negativo y otro que diga si un número es par o impar. La clase no debe tener ningún atributo. Se deben probar los métodos en el main.
3. Realizar un programa que calcule el volumen de un cilindro. Vosotros debéis decidir qué método usar y qué valores pasarle.
4. Crear un programa que simule el lanzamiento de una moneda (cara o cruz). Para ello, se usarán los siguientes métodos:
  - Generar número entero entre 1 y 2 (equivalente a lanzar la moneda). Devuelve el número entero generado.
  - Método para comprobar ganador, se le pasarán el número elegido por el jugador y dentro se generará el aleatorio llamando al método anterior. Devuelve true si son iguales o false si no.
  - Un método para mostrar en bonito el ganador. Si se le pasa true mostrar un mensaje de ganador y si no, debe mostrar "Siga jugando".
  - En el main, se debe crear lo necesario para simular un juego de cara o cruz.
5. Realizar un programa que lea un texto del teclado y un número entero y muestre en pantalla dicho texto una cantidad de veces igual al número introducido. La clase se puede llamar "Copiado".
6. Se desea llevar un control del estado de una cuenta corriente. Una cuenta corriente está caracterizada por su saldo y el nombre del titular y sobre ella se pueden realizar tres tipos de operaciones:
  - Ingresar (cantidad): ingresa en la cuenta una cantidad de dinero dada por el usuario. No imprime de nuevo el saldo, pero sí lo actualiza.
  - Retirar (cantidad): Debe comprobar si queda suficiente saldo y si es así, disminuye la cuenta una determinada cantidad de dinero especificada por el usuario. No imprime de nuevo el saldo, pero sí lo actualiza.
  - Calcular saldo en dólares americanos. Se le debe pasar el valor de cambio y debe devolver el valor del saldo en dólares.
  - Consultar el saldo.

Supón que la cuenta inicialmente tiene un saldo de cero. Escribe una clase CuentaCorriente que implemente las funcionalidades descritas y un main con un menú para probarlas.

7. Crear un programa que gestione un sorteo de lotería. Tendremos la clase Sorteo y la principal para probar todo. El programa debe hacer lo siguiente:

- Posibilidad de comprar/generar un décimo (elegir un número).
- Hacer sorteo, donde se generará aleatoriamente el número premiado.
- Comprobar si nuestro décimo está premiado.
- Mostrar si eres ganador o no (en el main).

**Nota: si alguien quiere "entrenar" más, se pueden realizar todos los ejercicios hechos hasta ahora en los temas 1 y 2 mediante programación orientada a objetos.**

8. Crear un programa que emule una máquina vendedora de tickets de metro. Tendremos que crear la clase Ticket (Clase POJO), la clase Maquina (donde estarán todos los métodos con funcionalidad) y la clase principal o de prueba. Podremos:

- Comprar uno o varios billetes.
- Calcular el cambio a devolver.
- Imprimir por pantalla el billete. Solo uno. Si se han comprado "x" debemos mostrar en el ticket el mensaje "válido para x personas".

Además, al final de la jornada, el operario que lleva el mantenimiento de la máquina, debe tener disponible un método que le ofrezca el saldo total de la recaudación de la máquina en ese día (no es necesario usar fechas) y ponga a cero el contador.

Por último, también debe contar con la posibilidad de cambiar el precio de los billetes ya que estos suelen subir todos los años.

Las operaciones para el operario se harán solo si se introduce la contraseña adecuada.

Estas últimas funcionalidades también se harán con métodos dentro de la clase Maquina.

9. Crear un programa para poner las notas de un alumno. El programa debe poder poner notas a un solo alumno, mostrar todas sus notas por pantalla, modificar una nota, calcular la media y dar su número de suspensos.

Se puede hacer guardando las notas en un array de doubles en lugar de una variable para cada nota. (Opcional) Se puede añadir al programa la posibilidad de usar el programa para varios alumnos, con lo que debemos usar un array de alumnos (array de objetos tipo Alumno).

10. 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 y una lista de localidades (tipo Entrada para no liar mucho el ejercicio). No es necesario usar arrays de Entradas, esa parte la haremos más adelante, en la segunda parte del tema.

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 AMPLIACIÓN (no se entregan ni se corregirán)

Realizar el ejercicio de la máquina expendedora con el siguiente diagrama de clases:

