

1º (A)

Desarrolla un sistema de gestión de carrito de compras para una tienda en línea. El sistema debe permitir a los usuarios agregar productos al carrito, calcular el total a pagar aplicando un descuento global del 10%, y mostrar los detalles de los productos en el carrito.

Productos Disponibles:

- Los productos deben pertenecer a una de las siguientes categorías: ELECTRONICA, ROPA, o ALIMENTACION.
- Cada producto tiene un nombre, un precio, y una categoría.

Carrito de Compras:

- El carrito tiene una capacidad fija de productos.
- Los usuarios pueden agregar productos al carrito siempre que no se supere la capacidad.
- El sistema debe calcular el precio total con un descuento global del 10% aplicado a cada producto.
- Debe mostrarse el nombre y el precio con descuento de cada producto, así como el total final del carrito.

Operaciones Permitidas:

- Agregar un nuevo producto al carrito.
- Mostrar los productos en el carrito junto con el total a pagar.

Clases Requeridas:

- CategoriaProducto: Una enumeración que define las categorías de los productos (ELECTRONICA, ROPA, ALIMENTACION).
- Producto: Clase que representa un producto, con los atributos nombre, precio, y categoria. Incluye un método para aplicar un descuento global del 10%.
- Carrito: Clase que maneja una colección de productos, permite agregar productos hasta un límite definido, y calcula el total a pagar con el descuento aplicado.
- Test: Clase principal que simula la creación de productos, su adición a un carrito, y la impresión de los detalles del carrito y el total final.

Resultado de ejecución:

Laptop - Precio final: 1080.0

Camiseta - Precio final: 18.0

Manzanas - Precio final: 2.25

Total: 1100.25

1º (B)

Crea una aplicación para gestionar una cuenta bancaria. Las cuentas pueden ser de diferentes tipos, como AHORRO, CORRIENTE, Y EMPRESARIAL. Además, los clientes pueden realizar transacciones (depósitos y retiros) y cada tipo de cuenta tiene una tasa de interés diferente.

Requisitos:

- Crea un tipo enumerado TipoCuenta con los valores AHORRO, CORRIENTE, EMPRESARIAL.
- Crea una clase CuentaBancaria con los atributos titular, saldo, tipoCuenta y un método realizarTransaccion para depósitos y retiros.

Los atributos titular y saldo deben ser finales.

- Implementa un método en la clase Banco para calcular el total de dinero en todas las cuentas utilizando un array de CuentaBancaria como parámetro.

La clase CuentaBancaria debe incluir un método final calcularIntereses que calcule los intereses de la cuenta según su tipo.

Resultado de ejecución:

Juan Pérez - Saldo: 5000,00€ - Intereses: 100,00€

Ana López - Saldo: 3000,00€ - Intereses: 30,00€

Carlos García - Saldo: 10000,00€ - Intereses: 500,00€

Total de dinero en el banco: 18000,00€

2º (B)

Crea un sistema de gestión de una biblioteca en el que los libros tengan categorías y se pueda calcular el valor total de todos los libros disponibles. Cada libro tiene un título, un precio y una categoría (FICCIÓN, CIENCIA, HISTORIA). Los libros pueden ser prestados, y si un libro está prestado, no se puede incluir en el cálculo del valor total de la biblioteca.

Requisitos:

- Crea un tipo enumerado `CategoriaLibro` con los valores FICCIÓN, CIENCIA, HISTORIA.
- La clase `Libro` tiene los atributos `titulo`, `precio`, `categoria` y `prestado` (booleano).
- La clase `Biblioteca` tiene un array de `Libro` y un método que calcule el valor total de todos los libros que no están prestados.
- La clase `Libro` debe tener un método `marcarPrestado()` para cambiar el estado de un libro a prestado.

1º (C)

Desarrolla un sistema de empleados en una empresa. Los empleados tienen un rol (ADMIN, GERENTE, EMPLEADO) y un salario. Debes calcular el salario final de cada empleado en función de su rol y un bono estático. La clase `Empleado` tiene métodos para calcular el salario global de todos los empleados, y una constante final para el (?) base.

Requisitos:

- Crea un tipo enumerado `Rol` con valores ADMIN, GERENTE, EMPLEADO.
- La clase `Empleado` debe tener los atributos `nombre`, `rol`, `salarioBase` y un atributo `BONO_BASE` que se aplica a todos los empleados.
- Implementa un método en la clase `Empresa` que calcule el salario total de todos los empleados usando un array de `Empleado` como parámetro.
- Los empleados deben ser gestionados en un array dentro de la clase `Empresa`.

Resultado de ejecución:

Juan:4500,00€

Ana:3000,00€

Carlos:2000,00€

Lucía:2100,00€

Total de salarios: 11600,00€

2º (C)

Desarrolla un sistema para gestionar el inventario de una tienda online.

Cada producto tiene un nombre, precio y cantidad en stock. Los productos pueden tener un descuento fijo (por ejemplo, 10%). Los descuentos se aplican solo a productos disponibles en inventario. Además, la tienda tiene un descuento global para todos los productos, que se aplica al precio después de los descuentos individuales.

Requisitos:

- Crea un tipo enumerado CategoriaProducto con valores como ELECTRONICA, ROPA, ALIMENTACION.
- Crea una clase Producto que tenga los atributos nombre, precio, cantidadEnStock y un atributo estático DESCUENTO_GLOBAL.
- Crea una clase Inventario que contenga un array de productos.
- Implementa un método que calcule el precio final de todos los productos considerando los descuentos y la cantidad en stock.

Resultado de ejecución:

Producto: Laptop - Precio final por unidad: 1200,00€ - Stock: 10

Producto: Camiseta - Precio final por unidad: 20,00€ - Stock: 50

Producto: Manzanas - Precio final por unidad: 2,50€ - Stock: 200

Total del inventario: 12150,00€