

Desarrolla una aplicación de consola en Java para gestionar los **productos** de una tienda online pero solo utilizando funciones lambda, métodos de acceso directo y bipredicate.

Cada **producto** debe tener: nombre (String), categoria (String), codigo (String, **único**), precio (double)

Los requisitos de la aplicación son los siguientes:

Clase Producto

- Atributos privados para todos los campos
- **Constructores:**
 - Vacío
 - Completo (todos los parámetros)
 - Sin precio (inicialízalo a 1)
- Método void mostrarDatos() que imprima toda la información
- **Getters y setters** completos

Clase GestorProductos

Gestiona una lista de productos con **estos métodos exactos**:

1. void agregarProducto(Producto p): Solo agrega si el código NO existe ya en la lista
2. void asignarPrecios(): Asigna precios aleatorios (1-100) a productos sin precio (precio < 0)
3. void mostrarProductos(): Muestra todos los productos de la lista
4. double calcularPrecioMedio(): Devuelve el precio medio de todos los productos
5. long getNumeroProductosCaros(double limite): Cuenta productos con precio \geq límite
6. List<Producto> getProductosCaros(double limite): Devuelve lista de productos con precio $>$ límite
7. Optional<Producto> buscarPorCodigo(String codigo): Devuelve el producto si existe (Optional vacío si no)
8. void ordenarPorPrecioDesc(): Ordena la lista por precio de mayor a menor
9. List<Producto> getProductosPorCondicion(BiPredicate<Producto,Double> condicion, double valor): Devuelve productos según el criterio pasado

Clase Main

- Crea un GestorProductos
- Agrega **varios productos** de prueba (algunos con precio, otros sin precio)
- **Busca productos** por código usando:
 - ifPresent() para mostrar datos si existe
 - ifPresentOrElse() para buscar/crear si no existe
- Asigna precios, ordena por precio descendente y muestra la lista
- Prueba getProductosPorCondicion() con **distintos criterios** de precio