INTEGRADOR FORMA 1

public class Articulo {

private String codigo;

private String descripcion;

private double precioVenta;

private int stock;

public Articulo(String codigo, String descripcion, double precioVenta, int stock) {

this.codigo = codigo;

this.descripcion = descripcion;

this.precioVenta = precioVenta;

this.stock = stock;

}

// Getters

public String getCodigo() { return codigo; }

public String getDescripcion() { return descripcion; }

public double getPrecioVenta() { return precioVenta; }

public int getStock() { return stock; }

// Setters

public void setCodigo(String codigo) { this.codigo = codigo; }

public void setDescripcion(String descripcion) { this.descripcion = descripcion; }

public void setPrecioVenta(double precioVenta) { this.precioVenta = precioVenta; }

public void setStock(int stock) { this.stock = stock; }

@Override

public String toString() {

return "Código: " + codigo + ", Descripción: " + descripcion + ", Precio de Venta: " + precioVenta + ", Stock: " + stock;

}

}

1. **Definición de la clase Almacen:**

java

Copiar código

import java.util.HashMap;

import java.util.Map;

import java.util.Scanner;

public class Almacen {

private Map<String, Articulo> inventario;

public Almacen() {

this.inventario = new HashMap<>();

}

public void agregarArticulo(Articulo articulo) {

inventario.put(articulo.getCodigo(), articulo);

}

public void actualizarStock(String codigo, int cantidad) {

Articulo articulo = inventario.get(codigo);

if (articulo != null) {

articulo.setStock(articulo.getStock() + cantidad);

} else {

System.out.println("Artículo no encontrado.");

}

}

public void actualizarPrecio(String codigo, double nuevoPrecio) {

Articulo articulo = inventario.get(codigo);

if (articulo != null) {

articulo.setPrecioVenta(nuevoPrecio);

} else {

System.out.println("Artículo no encontrado.");

}

}

public void listarArticulos() {

if (inventario.isEmpty()) {

System.out.println("No hay artículos en el almacén.");

} else {

for (Articulo articulo : inventario.values()) {

System.out.println(articulo);

}

}

}

public static void main(String[] args) {

Almacen almacen = new Almacen();

Scanner scanner = new Scanner(System.in);

while (true) {

System.out.println("\n1. Ingresar nuevo producto");

System.out.println("2. Agregar stock a producto existente");

System.out.println("3. Actualizar precio de producto");

System.out.println("4. Ver listado de artículos");

System.out.println("5. Salir");

System.out.print("Seleccione una opción: ");

int opcion = scanner.nextInt();

scanner.nextLine(); // Consumir el salto de línea

switch (opcion) {

case 1:

System.out.print("Ingrese el código del producto: ");

String codigo = scanner.nextLine();

System.out.print("Ingrese la descripción del producto: ");

String descripcion = scanner.nextLine();

System.out.print("Ingrese el precio de venta del producto: ");

double precioVenta = scanner.nextDouble();

System.out.print("Ingrese el stock inicial del producto: ");

int stock = scanner.nextInt();

scanner.nextLine(); // Consumir el salto de línea

Articulo nuevoArticulo = new Articulo(codigo, descripcion, precioVenta, stock);

almacen.agregarArticulo(nuevoArticulo);

System.out.println("Producto agregado exitosamente.");

break;

case 2:

System.out.print("Ingrese el código del producto: ");

String codigoStock = scanner.nextLine();

System.out.print("Ingrese la cantidad a agregar: ");

int cantidad = scanner.nextInt();

scanner.nextLine(); // Consumir el salto de línea

almacen.actualizarStock(codigoStock, cantidad);

System.out.println("Stock actualizado exitosamente.");

break;

case 3:

System.out.print("Ingrese el código del producto: ");

String codigoPrecio = scanner.nextLine();

System.out.print("Ingrese el nuevo precio: ");

double nuevoPrecio = scanner.nextDouble();

scanner.nextLine(); // Consumir el salto de línea

almacen.actualizarPrecio(codigoPrecio, nuevoPrecio);

System.out.println("Precio actualizado exitosamente.");

break;

case 4:

almacen.listarArticulos();

break;

case 5:

System.out.println("Saliendo del programa...");

scanner.close();

return;

default:

System.out.println("Opción no válida. Intente de nuevo.");

}

}

}

}

**Explicación del Código:**

1. **Clase Articulo:** Representa un artículo con atributos como código, descripción, precio de venta y stock. Incluye métodos getters, setters y el método toString.
2. **Clase Almacen:** Gestiona un inventario de artículos utilizando un HashMap para almacenar los artículos con sus códigos como clave. Contiene métodos para agregar artículos, actualizar stock, actualizar precios y listar artículos.
3. **Método main:** Permite al usuario interactuar con el almacén a través de la consola, ofreciendo opciones para ingresar nuevos productos, agregar stock, actualizar precios y ver el listado de artículos. La interacción se maneja mediante un bucle while que se ejecuta hasta que el usuario decide salir.

Este programa básico proporciona las funcionalidades necesarias para gestionar un almacén de productos a través de la consola.