

Examen Final - Programación en Java

Caso: Sistema de Gestión de Pedidos y Productos

Una tienda desea implementar un sistema sencillo para registrar productos y gestionar los pedidos que realizan los clientes. Cada pedido contiene un solo producto. El sistema debe permitir:

1. Registrar, modificar y eliminar productos.
 2. Registrar un pedido que contenga el producto y la fecha del pedido.
 3. Implementar una transacción al registrar el pedido para asegurar que:
 - El producto exista.
 - El pedido se registre correctamente.
 - Si ocurre algún error, el pedido no se registre.
-

Requisitos Técnicos

- Conexión a base de datos MySQL usando JDBC.
 - CRUD básico de productos usando JDBC.
 - Uso de Hibernate para mapear las clases **Producto** y **Pedido**.
 - Implementación de una transacción para registrar el pedido (puede ser con JDBC o Hibernate).
-

Base de Datos

Nombre: tienda_db

```
CREATE DATABASE tienda_db;
USE tienda_db;

CREATE TABLE productos (
  id INT AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR(100),
  precio DECIMAL(10,2)
);

CREATE TABLE pedidos (
  id INT AUTO_INCREMENT PRIMARY KEY,
  id_producto INT,
  fecha_pedido DATE,
  FOREIGN KEY (id_producto) REFERENCES productos(id)
);
```

Clases Recomendadas

1. Modelo (Hibernate):

- `Producto.java`
- `Pedido.java`

2. DAO JDBC:

- `ProductoDAO.java` → Métodos:
 - `insertarProducto()`
 - `actualizarProducto()`
 - `eliminarProducto()`
 - `listarProductos()`

3. Servicio:

- `PedidoService.java` → Método:
 - `registrarPedido(int idProducto):`
 - Validar si el producto existe.
 - Registrar el pedido.
 - Todo dentro de una **transacción**.

4. Clase Principal:

- `TiendaApp.java`
Opciones:
 1. Registrar producto
 2. Listar productos
 3. Registrar pedido
 4. Salir

Entregables

- Crear un documento Word que contenga:
 - Link del Proyecto Java completo: Puede ser un enlace a Google Drive (archivo .zip) o un repositorio público en GitHub.
 - Capturas de pantalla mostrando el funcionamiento del sistema.