

L1 Ejercicios de aplicación #1 Tecnologías de base de datos

Desarrollo

1. Decidir si utilizar una base de datos relacional o NoSQL y justificar la elección.

Elección: Base de datos relacional (MySQL)

El modelo tiene relaciones claras: clientes – pedidos - productos. Se requiere consistencia en transacciones (por ejemplo, al registrar un pedido con múltiples productos).

Ideal para consultas complejas con JOINS y agregaciones.

Escenario clásico de e-commerce: bien estructurado, con entidades normalizadas.

2. Diseñar un esquema básico de la base de datos. Si eliges una base de datos relacional, diseña las tablas y sus relaciones. Si eliges una base de datos NoSQL, define la estructura de los documentos o colecciones.

CLIENTES

- id_cliente (PK)
- nombre
- correo
- fecha_registro

PRODUCTOS

- id_producto (PK)
- nombre
- precio
- stock

PEDIDOS

- id_pedido (PK)
- id_cliente (FK)
- fecha_pedido
- estado

DETALLE_PEDIDO

- id_detalle (PK)
- id_pedido (FK)
- id_producto (FK)
- cantidad
- precio_unitario

Relaciones:

- Un cliente puede tener muchos pedidos - relación 1:N
 - Un pedido puede tener muchos productos - relación N:M
3. Implementar consultas básicas para obtener: Todos los pedidos de un cliente específico. Los productos de un pedido en particular. La cantidad total de pedidos realizados por cada

- a) Todos los pedidos de un cliente específico

```
SELECT p.id_pedido, p.fecha_pedido, p.estado
FROM PEDIDOS p
JOIN CLIENTES c ON p.id_cliente = c.id_cliente
WHERE c.nombre = 'luli';
```
- b) Productos de un pedido en particular

```
SELECT pr.nombre, dp.cantidad, dp.precio_unitario
FROM DETALLE_PEDIDO dp
JOIN PRODUCTOS pr ON dp.id_producto = pr.id_producto
WHERE dp.id_pedido = 101;
```
- c) Cantidad total de pedidos por cliente

```
SELECT c.nombre, COUNT(p.id_pedido) AS total_pedidos
FROM CLIENTES c
JOIN PEDIDOS p ON c.id_cliente = p.id_cliente
GROUP BY c.id_cliente;
```