Operaciones CRUD en el Sistema de Gestión de Pedidos

Introducción

El sistema actual gestiona clientes, cajones (conjuntos de productos) y pedidos. Cada operación CRUD se realizará sobre estas entidades principales utilizando PostgreSQL.

Operaciones Create (Inserción)

1. Inserción de Clientes

-- Ejemplo de inserción de un cliente

INSERT INTO Clientes (

numero\_documento,

nombre,

apellidos

) VALUES (

'12345678901', -- Número de documento (CI)

'Juan', -- Nombre

'Pérez González' -- Apellidos

);

-- Inserción de teléfonos del cliente

INSERT INTO Telefonos (

cliente\_id,

numero

) VALUES

((SELECT currval('clientes\_cliente\_id\_seq')), '095123456'),

((SELECT currval('clientes\_cliente\_id\_seq')), '096789012');

Características importantes:

- Se utiliza `currval()` para obtener el último ID generado

- Se valida el formato del documento (11 dígitos)

- Se permite múltiples teléfonos por cliente

2. Inserción de Cajones

-- Inserción de un cajón

INSERT INTO Cajones (

nombre,

descripcion

) VALUES (

'Cajón Básico',

'Selección de productos para consumo diario'

);

-- Agregar productos al cajón

INSERT INTO Productos\_Cajon (

cajon\_id,

producto\_id,

cantidad\_kg

) VALUES

((SELECT currval('cajones\_cajon\_id\_seq')), 1, 2.5),

((SELECT currval('cajones\_cajon\_id\_seq')), 2, 1.8);

Detalles:

- Se crea primero el cajón

- Luego se agregan los productos con sus cantidades

- Se verifica la existencia previa de productos

3. Inserción de Pedidos

-- Inserción de un pedido

INSERT INTO Pedidos (

cliente\_id,

fecha\_entrega,

forma\_pago

) VALUES (

1, -- ID del cliente

'2024-03-15', -- Fecha de entrega

'transferencia' -- Forma de pago

);

-- Detalle de pago para transferencia

INSERT INTO Detalle\_Pago (

pedido\_id,

cuenta\_origen

) VALUES (

currval('pedidos\_pedido\_id\_seq'),

'1234-5678-9012-3456'

);

-- Asociar cajones al pedido

INSERT INTO Pedidos\_Cajones (

pedido\_id,

cajon\_id

) VALUES

(currval('pedidos\_pedido\_id\_seq'), 1),

(currval('pedidos\_pedido\_id\_seq'), 2);

Características:

- Registro del pedido con sus detalles

- Información adicional para transferencias

- Asociación de cajones al pedido

Operaciones Read (Consulta)

1. Consulta de Clientes

SELECT

c.numero\_documento,

c.nombre,

c.apellidos,

string\_agg(t.numero, ', ') as telefonos,

COUNT(p.pedido\_id) as total\_pedidos

FROM

Clientes c

LEFT JOIN

Telefonos t ON c.cliente\_id = t.cliente\_id

LEFT JOIN

Pedidos p ON c.cliente\_id = p.cliente\_id

GROUP BY

c.cliente\_id, c.numero\_documento, c.nombre, c.apellidos

ORDER BY

c.apellidos, c.nombre;

2. Consulta de Cajones

SELECT

c.nombre,

c.descripcion,

COUNT(pc.producto\_id) as total\_productos,

SUM(pc.cantidad\_kg \* p.precio\_kg) as precio\_total

FROM

Cajones c

LEFT JOIN

Productos\_Cajon pc ON c.cajon\_id = pc.cajon\_id

LEFT JOIN

Productos p ON pc.producto\_id = p.producto\_id

GROUP BY

c.cajon\_id, c.nombre, c.descripcion

ORDER BY

c.nombre;

3. Consulta de Pedidos

SELECT

p.pedido\_id,

c.nombre || ' ' || c.apellidos as cliente,

p.fecha\_pedido,

p.fecha\_entrega,

p.forma\_pago,

string\_agg(ca.nombre, ', ') as cajones

FROM

Pedidos p

JOIN

Clientes c ON p.cliente\_id = c.cliente\_id

JOIN

Pedidos\_Cajones pc ON p.pedido\_id = pc.pedido\_id

JOIN

Cajones ca ON pc.cajon\_id = ca.cajon\_id

GROUP BY

p.pedido\_id, c.nombre, c.apellidos

ORDER BY

p.fecha\_pedido DESC;

Operaciones Update (Actualización)

1. Actualizar Información de Cliente

UPDATE Clientes

SET

nombre = 'Juan Carlos',

apellidos = 'Pérez Martínez'

WHERE

numero\_documento = '12345678901';

-- Actualizar teléfonos

UPDATE Telefonos

SET numero = '097654321'

WHERE

cliente\_id = 1 AND

numero = '095123456';

2. Modificar Cajón

UPDATE Cajones

SET

nombre = 'Cajón Premium',

descripcion = 'Selección de productos gourmet'

WHERE

cajon\_id = 1;

-- Actualizar productos del cajón

UPDATE Productos\_Cajon

SET cantidad\_kg = 3.0

WHERE

cajon\_id = 1 AND

producto\_id = 1;

Operaciones Delete (Eliminación)

1. Eliminar Cliente

-- Eliminar teléfonos primero (por restricciones de integridad)

DELETE FROM Telefonos

WHERE cliente\_id = 1;

-- Luego eliminar el cliente

DELETE FROM Clientes

WHERE cliente\_id = 1;

2. Eliminar Cajón

-- Eliminar productos asociados al cajón

DELETE FROM Productos\_Cajon

WHERE cajon\_id = 1;

-- Eliminar el cajón

DELETE FROM Cajones

WHERE cajon\_id = 1;

3. Eliminar Pedido

-- Eliminar detalles de pago

DELETE FROM Detalle\_Pago

WHERE pedido\_id = 1;

-- Eliminar asociaciones de cajones

DELETE FROM Pedidos\_Cajones

WHERE pedido\_id = 1;

-- Eliminar el pedido

DELETE FROM Pedidos

WHERE pedido\_id = 1;

Consideraciones Importantes

1. Siempre seguir el orden de eliminación respetando las restricciones de integridad referencial

2. Utilizar transacciones para operaciones complejas

3. Validar datos antes de realizar inserciones o actualizaciones

4. Manejar excepciones y realizar validaciones en el código C++

Esta guía proporciona una visión completa de las operaciones CRUD en el sistema de gestión de pedidos, mostrando cómo interactuar con la base de datos PostgreSQL de manera segura y eficiente.