Consultas SQL en el Sistema de Gestión de Taller Mecánico

Introducción a las Operaciones CRUD

En el contexto de nuestro sistema de gestión de taller mecánico, las operaciones CRUD (Create, Read, Update, Delete) son fundamentales para manejar la información de clientes, vehículos, reparaciones y mecánicos. Cada operación tiene un propósito específico que contribuye a la gestión integral del taller.

Operaciones Create (Inserción)

Inserción de Clientes

INSERT INTO Clientes (dni, nombre, apellidos, direccion, telefono)

VALUES ('12345678A', 'Juan', 'García Pérez', 'Calle Principal 10', '678901234')

Objetivo

Registrar un nuevo cliente en el sistema con información personal completa. Esta consulta permite:

- Almacenar datos únicos de identificación

- Garantizar la integridad de la información personal

- Preparar el registro para futuras asociaciones (como vehículos)

Inserción de Vehículos

INSERT INTO Vehiculos (matricula, modelo, color, dni\_cliente)

VALUES ('1234ABC', 'Ford Focus', 'Rojo', '12345678A')

Objetivo

Vincular un vehículo a un cliente específico, estableciendo una relación directa mediante:

- Matrícula única

- Información descriptiva del vehículo

- Referencia al cliente propietario

Inserción de Reparaciones

INSERT INTO Reparaciones (

matricula\_vehiculo,

fecha\_entrada,

hora\_entrada,

mecanico\_principal,

estado

) VALUES (

'1234ABC',

CURRENT\_DATE,

CURRENT\_TIME,

1,

'En evaluación'

)

Objetivo

Iniciar el proceso de reparación, capturando:

- Vehículo en reparación

- Timestamps de entrada

- Mecánico asignado

- Estado inicial del proceso

Operaciones Read (Consulta)

Consulta de Reparaciones por Cliente

SELECT

v.matricula,

v.modelo,

r.fecha\_entrada,

r.estado,

m.nombre AS nombre\_mecanico

FROM Reparaciones r

JOIN Vehiculos v ON r.matricula\_vehiculo = v.matricula

JOIN Clientes c ON v.dni\_cliente = c.dni

JOIN Mecanicos m ON r.mecanico\_principal = m.id\_mecanico

WHERE c.dni = '12345678A'

ORDER BY r.fecha\_entrada DESC

Objetivo

Recuperar un histórico completo de reparaciones de un cliente específico, proporcionando:

- Detalles de los vehículos

- Fechas de reparación

- Estados de las reparaciones

- Información del mecánico principal

Consulta de Estado de Mecánicos

SELECT

nombre,

apellidos,

disponible,

(

SELECT COUNT(\*)

FROM Reparaciones

WHERE mecanico\_principal = Mecanicos.id\_mecanico

AND estado != 'Finalizada'

) as reparaciones\_activas

FROM Mecanicos

ORDER BY disponible DESC, reparaciones\_activas DESC

Objetivo

Obtener un resumen del estado actual de los mecánicos, mostrando:

- Disponibilidad

- Número de reparaciones en curso

- Carga de trabajo actual

Operaciones Update (Actualización)

Actualización del Estado de Reparación

UPDATE Reparaciones

SET

estado = 'En progreso',

mano\_obra = 75.50

WHERE id\_reparacion = 123

Objetivo

Modificar el estado de una reparación y registrar el trabajo realizado, permitiendo:

- Seguimiento del progreso

- Registro de costos de mano de obra

- Actualización dinámica del proceso de reparación

Actualización de Disponibilidad de Mecánicos

UPDATE Mecanicos

SET disponible = false

WHERE id\_mecanico = 1

Objetivo

Gestionar la disponibilidad de los mecánicos, facilitando:

- Control de asignación de trabajo

- Prevención de sobrecarga

- Optimización de recursos humanos

Operaciones Delete (Eliminación)

Eliminación de Reparaciones Pendientes

DELETE FROM Reparaciones

WHERE estado = 'Cancelada'

AND fecha\_entrada < CURRENT\_DATE - INTERVAL '30 days'

Objetivo

Mantener la base de datos limpia y actualizada mediante:

- Eliminación de registros antiguos

- Gestión del histórico de reparaciones

- Optimización del espacio de almacenamiento

Consultas Avanzadas

Generación de Factura

SELECT

c.nombre,

c.apellidos,

v.modelo,

v.matricula,

r.mano\_obra,

SUM(rr.cantidad \* rep.precio\_unidad) AS total\_repuestos,

r.mano\_obra + SUM(rr.cantidad \* rep.precio\_unidad) AS total\_reparacion

FROM Reparaciones r

JOIN Vehiculos v ON r.matricula\_vehiculo = v.matricula

JOIN Clientes c ON v.dni\_cliente = c.dni

LEFT JOIN Repuestos\_Reparacion rr ON r.id\_reparacion = rr.id\_reparacion

LEFT JOIN Repuestos rep ON rr.id\_repuesto = rep.id\_repuesto

WHERE r.id\_reparacion = 123

GROUP BY

c.nombre,

c.apellidos,

v.modelo,

v.matricula,

r.mano\_obra

Objetivo

Consolidar información para facturación, integrando:

- Datos del cliente

- Información del vehículo

- Costos de mano de obra

- Detalle de repuestos utilizados

- Cálculo de total de reparación

Consideraciones Finales

Las consultas SQL presentadas demuestran la versatilidad y potencia del diseño de base de datos para el sistema de gestión de taller mecánico. Cada operación está diseñada para:

1. Mantener la integridad de los datos

2. Proporcionar flexibilidad en la gestión

3. Facilitar el seguimiento de procesos

4. Optimizar la toma de decisiones

La estructura permite un manejo comprehensivo de la información, desde el registro inicial hasta la generación de informes y facturación.