

Projeto de Banco de dados lógico de oficina mecânica
Autor: Otoniele Santos
Data:03.04.25

-- Banco de Dados para Oficina Mecânica

CREATE DATABASE OficinaMecanica; USE OficinaMecanica;

-- Tabela de Clientes CREATE TABLE Clientes (id_cliente INT PRIMARY KEY
AUTO_INCREMENT, nome VARCHAR(100) NOT NULL, telefone VARCHAR(15), email
VARCHAR(100), endereco TEXT);

-- Tabela de Veículos CREATE TABLE Veiculos (id_veiculo INT PRIMARY KEY
AUTO_INCREMENT, id_cliente INT, marca VARCHAR(50), modelo VARCHAR(50), ano INT,
placa VARCHAR(10) UNIQUE, FOREIGN KEY (id_cliente) REFERENCES
Clientes(id_cliente));

-- Tabela de Mecânicos CREATE TABLE Mecanicos (id_mecanico INT PRIMARY KEY
AUTO_INCREMENT, nome VARCHAR(100) NOT NULL, especialidade VARCHAR(100));

-- Tabela de Ordens de Serviço (OS) CREATE TABLE OrdensServico (id_os INT PRIMARY
KEY AUTO_INCREMENT, id_veiculo INT, id_mecanico INT, data_emissao DATE NOT
NULL, data_conclusao DATE, valor_total DECIMAL(10,2), status VARCHAR(50), FOREIGN
KEY (id_veiculo) REFERENCES Veiculos(id_veiculo), FOREIGN KEY (id_mecanico)
REFERENCES Mecanicos(id_mecanico));

-- Tabela de Serviços CREATE TABLE Servicos (id_servico INT PRIMARY KEY
AUTO_INCREMENT, descricao VARCHAR(255) NOT NULL, preco DECIMAL(10,2) NOT
NULL);

-- Tabela de Peças CREATE TABLE Pecas (id_peca INT PRIMARY KEY
AUTO_INCREMENT, descricao VARCHAR(255) NOT NULL, preco DECIMAL(10,2) NOT
NULL);

-- Tabela de Serviços por OS CREATE TABLE OS_Servicos (id_os INT, id_servico INT,
quantidade INT DEFAULT 1, PRIMARY KEY (id_os, id_servico), FOREIGN KEY (id_os)
REFERENCES OrdensServico(id_os), FOREIGN KEY (id_servico) REFERENCES
Servicos(id_servico));

-- Tabela de Peças por OS CREATE TABLE OS_Pecas (id_os INT, id_peca INT, quantidade
INT DEFAULT 1, PRIMARY KEY (id_os, id_peca), FOREIGN KEY (id_os) REFERENCES
OrdensServico(id_os), FOREIGN KEY (id_peca) REFERENCES Pecas(id_peca));

-- Tabela de Pagamentos CREATE TABLE Pagamentos (id_pagamento INT PRIMARY KEY
AUTO_INCREMENT, id_os INT, valor DECIMAL(10,2) NOT NULL, metodo_pagamento
VARCHAR(50), data_pagamento DATE NOT NULL, FOREIGN KEY (id_os) REFERENCES
OrdensServico(id_os));

-- Queries Implementadas

-- Consulta de ordens de serviço por cliente e status SELECT OS.id_os, C.nome AS Cliente, V.modelo AS Veiculo, OS.status, OS.valor_total FROM OrdensServico OS JOIN Veiculos V ON OS.id_veiculo = V.id_veiculo JOIN Clientes C ON V.id_cliente = C.id_cliente WHERE OS.status = 'Em andamento';

-- Listagem de serviços realizados por mecânico SELECT M.nome AS Mecanico, S.descricao AS Servico, COUNT(OS_S.id_servico) AS Quantidade FROM OrdensServico OS JOIN Mecanicos M ON OS.id_mecanico = M.id_mecanico JOIN OS_Servicos OS_S ON OS.id_os = OS_S.id_os JOIN Servicos S ON OS_S.id_servico = S.id_servico GROUP BY M.nome, S.descricao;

-- Cálculo do total gasto por cliente SELECT C.nome AS Cliente, SUM(OS.valor_total) AS TotalGasto FROM OrdensServico OS JOIN Veiculos V ON OS.id_veiculo = V.id_veiculo JOIN Clientes C ON V.id_cliente = C.id_cliente GROUP BY C.nome;

-- Consulta de serviços mais frequentes SELECT S.descricao AS Servico, COUNT(OS_S.id_servico) AS Quantidade FROM OS_Servicos OS_S JOIN Servicos S ON OS_S.id_servico = S.id_servico GROUP BY S.descricao ORDER BY Quantidade DESC;

-- Filtragem e ordenação de OS por data e valor SELECT * FROM OrdensServico ORDER BY data_emissao DESC, valor_total DESC;

-- Relatório de faturamento da oficina por período SELECT YEAR(data_conclusao) AS Ano, MONTH(data_conclusao) AS Mes, SUM(valor_total) AS Faturamento FROM OrdensServico WHERE status = 'Concluído' GROUP BY YEAR(data_conclusao), MONTH(data_conclusao) ORDER BY Ano DESC, Mes DESC;