Trabalho da Disciplina de Projeto de Bancos de Dados

Prof. Fred Lucena

PROJETO DE BANCO DE DADOS RELACIONAL

Equipe:

|  |
| --- |
| Bruno Cezar |
|  |
| Bruno Henrique |  |
| José Elias |  |
|  |
| Juscelino Mamede |  |
|  |
| Rafaela Oliveira |  |
|  |

**OBJETIVO**

O objetivo é o registro das vendas e coleta das informações úteis sobre o consumo de seus produtos e serviços, a analise tendências de vendas e prever vendas futuras, identificar oportunidades de negócios e tomar decisões informadas sobre preços e estoque. Ao analisar os dados registrados, o posto de gasolina pode identificar quais produtos são mais populares entre seus clientes e ajustar sua oferta de acordo, melhorando seus produtos e serviços para atender às necessidades e preferências dos clientes e, consequentemente, aumentar suas receitas e rentabilidade.

**MODELO DE NEGOCIO**

O posto de gasolina pode registrar todas as vendas de combustível e outros produtos, coletando informações sobre o tipo de combustível, a quantidade vendida, o preço e o horário da venda. Com essas informações, o posto de gasolina pode criar um banco de dados para analisar as tendências de vendas, identificar quais produtos são mais populares e ajustar seus preços ou estoque de acordo.

**LEVANTAMENTO DE REQUISITOS**

1. **Identificação do cliente:** O registro deve incluir informações básicas sobre o cliente, como nome, endereço e número de telefone. Definir se ele é pessoa física ou Jurídica.
2. **Tipo de combustível:** O registro deve identificar o tipo de combustível comprado pelo cliente (gasolina, diesel, álcool, etc.)
3. **Quantidade de combustível:** O registro deve registrar a quantidade de combustível comprada pelo cliente, em litros ou em valor monetário.
4. **Preço do combustível:** O registro deve registrar o preço do combustível no momento da venda.
5. **Data e horário da venda:** O registro deve registrar a hora e a data da venda, para análise de tendências e planejamento de estoque.
6. **Forma de pagamento:** O registro deve identificar a forma de pagamento utilizada pelo cliente (dinheiro, cartão de crédito, débito, etc.)

**SGDB**

Utilizamos o SQL SERVER STUDIO.

**MINIMUNDO**

Um posto de gasolina precisa gerenciar informações sobre seus clientes, funcionários e produtos.

A entidade **Produto a**rmazena informações sobre os produtos disponíveis para venda, como o **ID do produto, nome, estoque e preço.**

A entidade **Veiculo** registra os veículos disponíveis, incluindo o **ID do veículo, modelo, cor, marca e placa**.

A entidade **Função** contém as diferentes funções dos funcionários, identificadas pelo **ID da função e pelo nome da função**.

A entidade **Cliente** armazena informações dos clientes, como **ID do cliente, telefone e e-mail.** Eles poderão ser pessoa **Física ou Jurídica.** os clientes que são **pessoas físicas**, com base no **CPF, ID do cliente, nome e data de nascimento**. os clientes que são **pessoas jurídicas,** utilizando o **CNPJ, ID do cliente, razão social da empresa.**

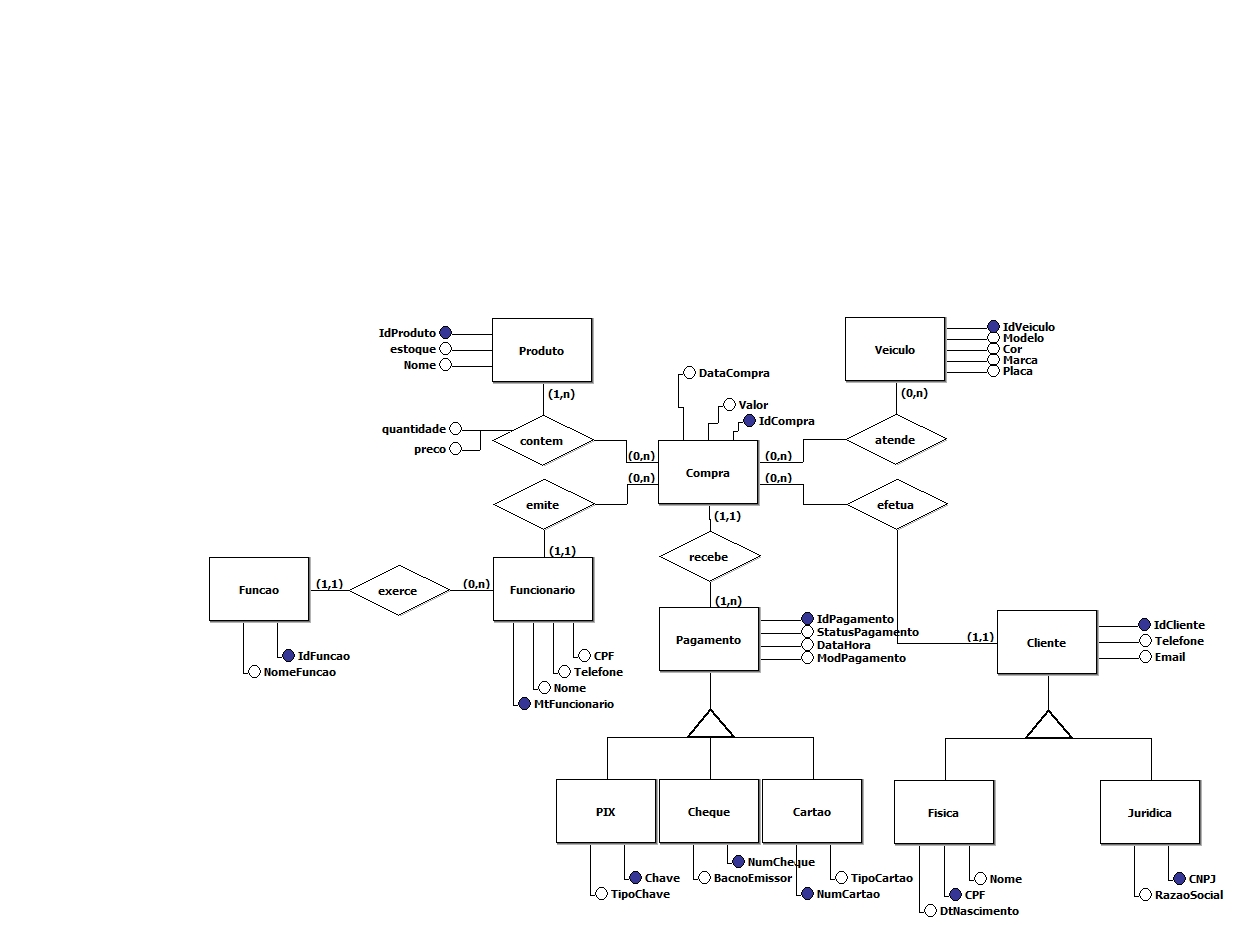
A entidade **Funcionário** contém informações dos funcionários, incluindo o número de matrícula do funcionário, **ID da função, CPF, telefone e nome.**

A entidade **Compra** registra as compras realizadas pelos clientes, com base na **ID da compra, número de matrícula do funcionário responsável, ID do cliente, valor da compra e data e hora da compra**.

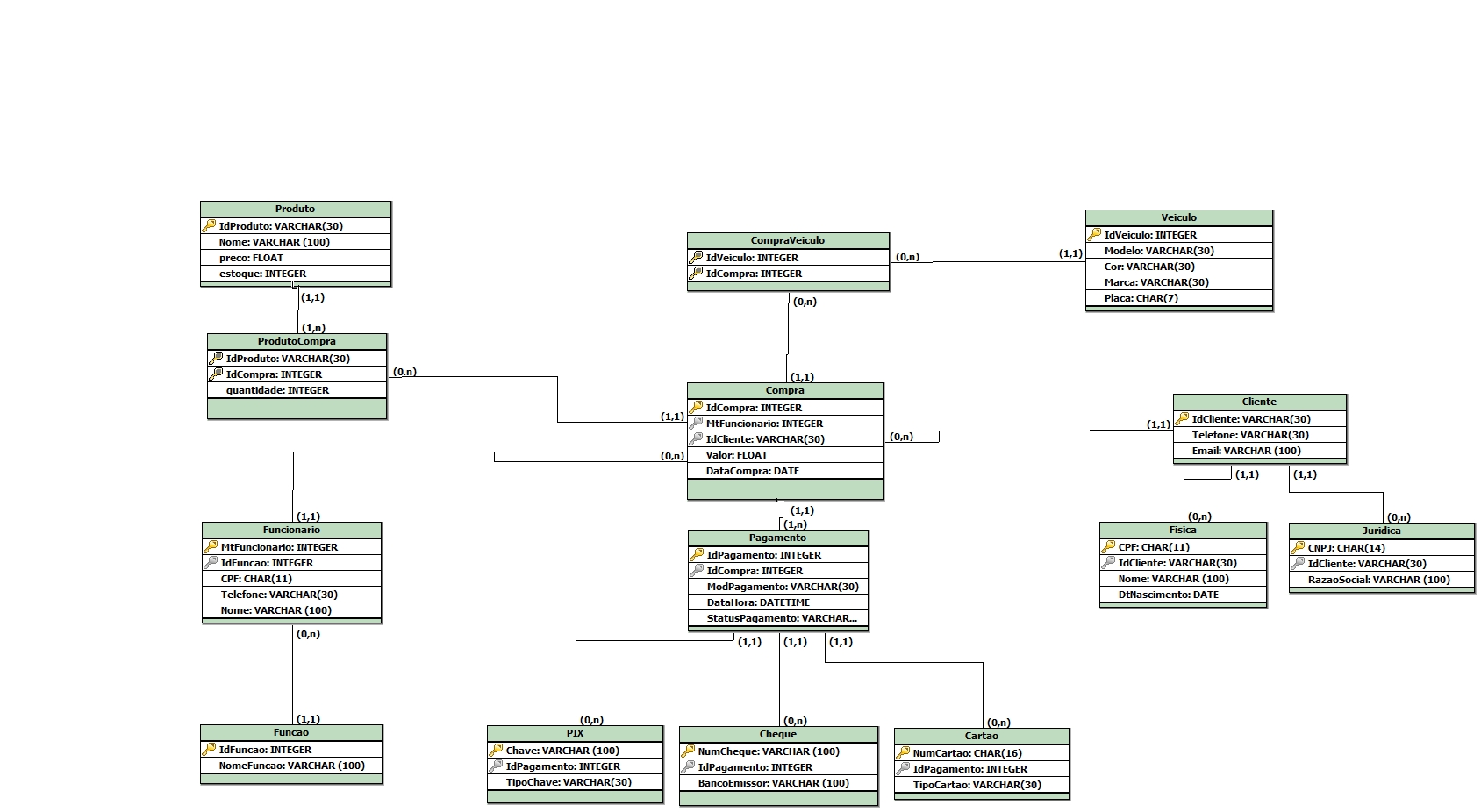
Tabela **Pagamento** registra os pagamentos efetuados pelas compras, utilizando o **ID do pagamento, ID da compra, modalidade de pagamento, data e hora do pagamento e status de pagamento (pago ou pendente)**. Os pagamentos podem tem sua Modalidade. E terão especialização para cartão **cheque e PIX.**

O modelo de banco de dados permitirá que o posto de gasolina mantenha um registro completo de seus clientes, veículos, transações de abastecimento e vendas de produtos, facilitando a análise e o planejamento de negócios.

**MODELO CONCEITUAL POSTO GASOLINA**



**MODELO LÓGICO POSTO GASOLINA**



**MODELO FÍSICO POSTO GASOLINA**

--\*\*\*\*\*\*\*\*\*\*\*\*\*\*CRIAÇÃO DO BANCO DE DADOS\*\*\*\*\*\*\*\*\*\*\*\*

CREATE DATABASE PostoGas;

go

USE PostoGas;

--\*\*\*\*\*\*\*\*\*\*\*\*\*\*CRIAÇÃO DAS TABELAS SEM FK\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

CREATE TABLE Produto(IdProduto INTEGER PRIMARY KEY IDENTITY,

                     nome      VARCHAR(100) NOT NULL,

                     preco     Float        NOT NULL,

                     estoque   INTEGER      NOT NULL,

                     CONSTRAINT Check\_PrecoPositivo CHECK (preco > 0));--clausula CHECK

CREATE TABLE Veiculo (IdVeiculo INTEGER IDENTITY PRIMARY KEY NOT NULL,

                      modelo    VARCHAR(30) NOT NULL,

                      cor       VARCHAR(30) NOT NULL,

                      marca     VARCHAR(30) NOT NULL,

                      placa     CHAR(7) NOT NULL,

                      CONSTRAINT Unica\_Placa UNIQUE (placa));--clausula UNIQUE

CREATE TABLE Funcao (IdFuncao   INTEGER PRIMARY KEY NOT NULL,

                     nomeFuncao VARCHAR(100) NOT NULL);

--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CRIAÇÃO DAS TABELAS COM FK\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

CREATE TABLE Cliente (IdCliente VARCHAR(30) PRIMARY KEY NOT NULL,

                      Telefone  VARCHAR(30),

                      Email     VARCHAR (100) NOT NULL);

CREATE TABLE Fisica (CPF        CHAR(11) PRIMARY KEY NOT NULL,

                     IdCliente  VARCHAR(30) NOT NULL,

                     Nome       VARCHAR (100) NOT NULL,

                     DtNascimento DATE,

                     FOREIGN KEY(IdCliente) REFERENCES Cliente (IdCliente));

CREATE TABLE Juridica (CNPJ        CHAR(14) PRIMARY KEY NOT NULL,

                       IdCliente   VARCHAR(30) NOT NULL,

                       RazaoSocial VARCHAR (100) NOT NULL,

                       FOREIGN KEY(IdCliente) REFERENCES Cliente (IdCliente));

CREATE TABLE Funcionario (MtFuncionario INTEGER IDENTITY (1000,10) PRIMARY KEY NOT NULL,

                          IdFuncao      INTEGER NOT NULL,

                          CPF           CHAR(11) NOT NULL,

                          Telefone      VARCHAR(30) DEFAULT '(081)9999-9999'NOT NULL,

                          Nome          VARCHAR (100) NOT NULL,

                          FOREIGN KEY(IdFuncao) REFERENCES Funcao (IdFuncao));

CREATE TABLE Compra (IdCompra        INTEGER IDENTITY PRIMARY KEY NOT NULL,

                     MtFuncionario   INTEGER ,

                     IdCliente       VARCHAR(30) NOT NULL,

                     Valor           FLOAT NOT NULL,

                     DataCompra      DATE NOT NULL,

                     StatusCompra    VARCHAR(30) NOT NULL,

                     FOREIGN KEY(MtFuncionario) REFERENCES Funcionario (MtFuncionario),

                     FOREIGN KEY(IdCliente) REFERENCES Cliente (IdCliente));

CREATE TABLE CompraVeiculo (IdVeiculo INTEGER NOT NULL,

                            IdCompra  INTEGER NOT NULL,

                            PRIMARY KEY (IdVeiculo, IdCompra),

                            FOREIGN KEY (IdVeiculo) REFERENCES Veiculo (IdVeiculo),

                            FOREIGN KEY (IdCompra) REFERENCES Compra (IdCompra));

CREATE TABLE ProdutoCompra (IdProduto   INTEGER NOT NULL,

                            IdCompra    INTEGER NOT NULL,

                            quantidade  INTEGER NOT NULL,

                            FOREIGN KEY(IdProduto) REFERENCES Produto (IdProduto),

                            FOREIGN KEY(IdCompra) REFERENCES Compra (IdCompra));

CREATE TABLE Pagamento (IdPagamento     INTEGER PRIMARY KEY NOT NULL,

                        IdCompra        INTEGER NOT NULL,

                        ModPagamento    VARCHAR(30) NOT NULL,

                        StatusPagamento VARCHAR(30) ,

                        DataHora        DATETIME NOT NULL,

                        FOREIGN KEY(IdCompra) REFERENCES Compra(IdCompra) ON DELETE CASCADE);

CREATE TABLE Cheque (NumCheque    VARCHAR (100) PRIMARY KEY NOT NULL,

                     IdPagamento  INTEGER NOT NULL,

                     BancoEmissor VARCHAR (100) NOT NULL,

                     FOREIGN KEY(IdPagamento) REFERENCES Pagamento (IdPagamento));

CREATE TABLE PIX (Chave       VARCHAR (100) PRIMARY KEY NOT NULL,

                  IdPagamento INTEGER NOT NULL,

                  TipoChave   VARCHAR(30) NOT NULL,

                  FOREIGN KEY(IdPagamento) REFERENCES Pagamento (IdPagamento));

CREATE TABLE Cartao (NumCartao   CHAR(16) PRIMARY KEY NOT NULL,

                     IdPagamento INTEGER,

                     TipoCartao  VARCHAR(30) NOT NULL,

                     FOREIGN KEY(IdPagamento) REFERENCES Pagamento (IdPagamento));

--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Alimentação das Tabelas\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

INSERT INTO Produto

      (Nome, preco, estoque)

VALUES('Gasolina', 4.99, 1000),

      ('Alcool',3.99, 500),

      ('Diesel',5.99, 800),

      ('Refrigerante Lata', 4.99, 200),

      ('Óleo Motor', 29.99, 50);

INSERT INTO Veiculo

        (Modelo, Cor, Marca, Placa)

VALUES  ('Uno', 'Branco', 'FIAT', 'ABC1234'),

        ('Palio', 'Prata', 'FIAT', 'DEF5678'),

        ('Onix', 'Preto', 'CHEVROLET', 'GHI9012'),

        ('Cruze', 'Vermelho', 'CHEVROLET', 'JKL3456'),

        ('S10', 'Azul', 'CHEVROLET', 'MNO7890');

INSERT INTO Funcao

        (IdFuncao, NomeFuncao)

VALUES  (1, 'Gerente'),

        (2, 'Vendedor'),

        (3, 'Caixa'),

        (4, 'Borracheiro'),

        (5, 'Frentista');

INSERT INTO Cliente

        (IdCliente, Telefone, Email)

VALUES  ('CLI01', '081998564785', 'cliente1@gmail.com'),

        ('CLI02', '081998562351', 'cliente2@gmail.com'),

        ('CLI03', '081998563652', 'cliente3@gmail.com'),

        ('CLI04', '081998562104', 'cliente4@gmail.com'),

        ('CLI05', '081998568647', 'cliente5@gmail.com'),

        ('CLI06', '081998564724', 'cliente6@TechNovaSolutions.com'),

        ('CLI07', '081998562302', 'cliente7@GlobalInnovations'),

        ('CLI08', '081998563637', 'cliente8@StarlightEnterprises.com'),

        ('CLI09', '081998562164', 'cliente9@Elite ServicesGroup.com'),

        ('CLI10', '081998568620', 'cliente10@PrimeTechIndustries.com');

INSERT INTO Fisica

       (CPF, IdCliente, Nome, DtNascimento)

VALUES ('12345678901', 'CLI01', 'Juscelino da Silva', CONVERT(date, '15/05/1990', 105)),

       ('23456789012', 'CLI02', 'Rafaela Oliveira', CONVERT(date, '20/08/1985', 105)),

       ('34567890123', 'CLI03', 'Bruno Cezar', CONVERT(date, '10/03/1995', 105)),

       ('45678901234', 'CLI04', 'Bruno Henrique', CONVERT(date, '25/11/1988', 105)),

       ('56789012345', 'CLI05', 'José Elias', CONVERT(date, '05/06/1992', 105));

INSERT INTO Juridica

        (CNPJ, IdCliente, RazaoSocial)

VALUES  ('56789123401234', 'CLI06', 'TechNova Solutions'),

        ('90123452345678', 'CLI07', 'Global Innovations'),

        ('01234563456789', 'CLI08', 'Starlight Enterprises'),

        ('89012344567567', 'CLI09', 'Elite Services Group'),

        ('89234015675678', 'CLI10', 'PrimeTech Industries');

INSERT INTO Funcionario

        (IdFuncao, CPF, Telefone, Nome)

VALUES  (1, '12345678901', '081998560102', 'Mario Robert'),

        (2, '23456789012', '081998560203', 'Robert Mario'),

        (3, '34567890123', '0819985632123', 'Maria Ferreira'),

        (4, '45678901234', '081998565262', 'Sulamita pink'),

        (5, '56789012345', '081998564847', 'Elder figueira');

INSERT INTO Compra

        (MtFuncionario, IdCliente, Valor, DataCompra, StatusCompra)

VALUES  (1040, 'CLI01', 100.00, CONVERT(date, '18-05-2023', 105),'Pago'),

        (1040, 'CLI02', 50.00, CONVERT(date, '18-05-2023', 105),'Pago'),

        (1010, 'CLI03', 75.00, CONVERT(date, '18-05-2023', 105),'Pago'),

        (1040, 'CLI04', 120.00, CONVERT(date, '18-05-2023', 105),'Pago'),

        (1040, 'CLI05', 200.00, CONVERT(date, '18-05-2023', 105),'Pago'),

        (1000, 'CLI06', 80.00, CONVERT(date, '18-05-2023', 105),'Pago'),

        (1020, 'CLI02', 150.00, CONVERT(date, '18-05-2023', 105),'Pago'),

        (1020, 'CLI03', 90.00, CONVERT(date, '18-05-2023', 105),'Pago'),

        (1020, 'CLI04', 180.00, CONVERT(date, '19-05-2023', 105),'Pago'),

        (1040, 'CLI05', 120.00, CONVERT(date, '19-05-2023', 105),'Pago'),

        (1040, 'CLI01', 150.50, CONVERT(date, '19-05-2023', 105),'Pago'),

        (1040, 'CLI06', 112.75, CONVERT(date, '19-05-2023', 105),'Pago'),

        (1010, 'CLI09', 250.30, CONVERT(date, '19-05-2023', 105),'Pago'),

        (1040, 'CLI03', 180.90, CONVERT(date, '19-05-2023', 105),'Pago'),

        (1040, 'CLI08', 350.20, CONVERT(date, '19-05-2023', 105),'Pago'),

        (1000, 'CLI04', 420.75, CONVERT(date, '20-05-2023', 105),'Pago'),

        (1020, 'CLI10', 180.60, CONVERT(date, '20-05-2023', 105),'Pago'),

        (1020, 'CLI05', 280.80, CONVERT(date, '20-05-2023', 105),'Pago'),

        (1020, 'CLI02', 150.10, CONVERT(date, '20-05-2023', 105),'Pago'),

        (1040, 'CLI07', 290.40, CONVERT(date, '20-05-2023', 105),'Pago');

INSERT INTO CompraVeiculo

        (IdVeiculo, IdCompra)

VALUES  (2, 1),

        (2, 2),

        (5, 3),

        (3, 4),

        (1, 5),

        (2, 6),

        (2, 7),

        (3, 8),

        (1, 9),

        (2, 10),

        (2, 11),

        (3, 12),

        (1, 13),

        (2, 14),

        (2, 15),

        (2, 16),

        (3, 17),

        (1, 18),

        (5, 19),

        (1, 20);

select \* from ProdutoCompra;

select \* from Produto;

INSERT INTO ProdutoCompra

        (IdProduto, IdCompra, quantidade)

VALUES  (1, 2, 24);

        (5, 3, 4),

        (1, 4, 40),

        (2, 5, 20),

        (1, 6, 30),

        (1, 7, 18),

        (1, 8, 36),

        (1, 9, 30),

        (1, 10, 30),

        (3, 11, 18),

        (3, 12, 42),

        (3, 13, 30),

        (5, 14, 12),

        (4, 15, 4),

        (2, 16, 45),

        (2, 17, 70),

        (3, 18, 37),

        (2, 19, 72),

        (1, 20, 20);

INSERT INTO Pagamento

        (IdPagamento, IdCompra, ModPagamento, DataHora,StatusPagamento)

VALUES  (1, 1, 'Cartão', CONVERT(datetime, '2023-05-18 10:00:00', 120),'Confirmado'),

        (2, 2, 'PIX', CONVERT(datetime, '2023-05-18 10:30:00', 120),'Confirmado'),

        (3, 3, 'Cheque', CONVERT(datetime, '2023-05-18 11:00:00', 120),'Confirmado'),

        (4, 4, 'Dinheiro', CONVERT(datetime, '2023-05-18 11:30:00', 120),'Confirmado'),

        (5, 5, 'Cartão', CONVERT(datetime, '2023-05-18 12:00:00', 120),'Confirmado'),

        (6, 6, 'PIX', CONVERT(datetime, '2023-05-18 12:30:00', 120),'Confirmado'),

        (7, 7, 'Cheque', CONVERT(datetime, '2023-05-18 13:00:00', 120),'Confirmado'),

        (8, 8, 'Dinheiro', CONVERT(datetime, '2023-05-18 13:30:00', 120),'Confirmado'),

        (9, 9, 'Cartão', CONVERT(datetime, '2023-05-19 14:00:00', 120),'Confirmado'),

        (10, 10, 'PIX', CONVERT(datetime, '2023-05-19 14:30:00', 120),'Confirmado'),

        (11, 11, 'Cartão', CONVERT(datetime, '2023-05-19 15:00:00', 120),'Confirmado'),

        (12, 12, 'PIX', CONVERT(datetime, '2023-05-19 15:30:00', 120),'Confirmado'),

        (13, 13, 'Cheque', CONVERT(datetime, '2023-05-19 16:00:00', 120),'Confirmado'),

        (14, 14, 'Dinheiro', CONVERT(datetime, '2023-05-19 16:30:00', 120),'Confirmado'),

        (15, 15, 'Cartão', CONVERT(datetime, '2023-05-19 17:00:00', 120),'Confirmado'),

        (16, 16, 'PIX', CONVERT(datetime, '2023-05-20 17:30:00', 120),'Confirmado'),

        (17, 17, 'Cheque', CONVERT(datetime, '2023-05-20 18:00:00', 120),'Confirmado'),

        (18, 18, 'Dinheiro', CONVERT(datetime, '2023-05-20 18:30:00', 120),'Confirmado'),

        (19, 19, 'Cartão', CONVERT(datetime, '2023-05-20 19:00:00', 120),'Confirmado'),

        (20, 20, 'PIX', CONVERT(datetime, '2023-05-20 19:30:00', 120),'Confirmado');

INSERT INTO Cartao

        (NumCartao, IdPagamento, TipoCartao)

VALUES  ('1122334455667788', 1, 'Crédito'),

        ('8877665544332211', 5, 'Débito'),

        ('2233115566448877', 9, 'Crédito'),

        ('1122445566337788', 11, 'Débito'),

        ('6655334411227700', 15, 'Crédito'),

        ('3344116655227700', 19, 'Crédito');

INSERT INTO PIX

        (Chave, IdPagamento, TipoChave)

VALUES  ('12345678901', 2, 'CPF'),

        ('12345678901234', 6, 'CNPJ'),

        ('cliente1@gmail.com', 10, 'E-mail'),

        ('081998563652', 16, 'Telefone'),

        ('PIX213sdf2151df156656', 20, 'Chave Aleatória');

INSERT INTO Cheque

        (NumCheque, IdPagamento, BancoEmissor)

VALUES  ('123654789', 3, 'Banco INTER'),

        ('987456321', 7, 'Banco Banco do Brasil'),

        ('456321789', 13, 'Caixa'),

        ('321654987', 17, 'Bradesco');

--\*\*\*\*\*\*\*\*\*\*Consultas\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

SELECT C.IdCompra,

       CL.IdCliente,

       F.Nome,

       V.Modelo,

       J.RazaoSocial

FROM

       Compra AS C

LEFT JOIN Cliente AS CL ON C.IdCliente = CL.IdCliente

LEFT JOIN Fisica AS F ON CL.IdCliente = F.IdCliente

LEFT JOIN Juridica AS J ON CL.IdCliente = J.IdCliente

LEFT JOIN CompraVeiculo AS CV ON C.IdCompra = CV.IdCompra

LEFT JOIN Veiculo AS V ON CV.IdVeiculo = V.IdVeiculo;

SELECT SUM(valor) AS TotalGastoPelocliente From Compra Where IdCliente = 'CLI02';

SELECT COUNT(\*) AS TotalComprasPelocliente FROM Compra Where IdCliente = 'CLI02';

SELECT AVG(valor) AS MediaGastoPelocliente FROM Compra Where IdCliente = 'CLI02';

SELECT MAX(valor) AS MaiorValorPelocliente FROM Compra Where IdCliente = 'CLI02';

SELECT MIN(valor) AS MenorValorPelocliente FROM Compra Where IdCliente = 'CLI02';

--\*\*\*\*\*\*\*\*\*\*Automação e Desempenho\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

--TRIGGER

CREATE TRIGGER trg\_novoEstoque

ON ProdutoCompra

FOR INSERT

AS

    BEGIN

        DECLARE @IdProduto INT,

                @estoque INT,

                @qtdVendida INT

        SELECT @IdProduto = IdProduto, @estoque = estoque FROM Produto

        SELECT @IdProduto = IdProduto, @qtdVendida = quantidade from inserted

        UPDATE Produto

        SET estoque = @estoque - @qtdVendida

        where idproduto = @IdProduto

    END

--STORED PROCEDURE

CREATE PROCEDURE Proc\_ListarVendas

    @DataVenda DATE

AS

    BEGIN

        SET @DataVenda = CONVERT(date, @DataVenda, 105)

        SELECT \*FROM Compra WHERE DataCompra = @DataVenda

    END

EXEC Proc\_ListarVendas '2023-05-18'

--FUNCTIONS

CREATE FUNCTION VendaFunci()

    RETURNS INT

AS

    BEGIN

        DECLARE @TotalVendas INT;

        SELECT @TotalVendas = COUNT(\*)

        FROM Compra

        WHERE MtFuncionario = 1040;

        RETURN @TotalVendas;

    END;

SELECT dbo.VendaFunci() AS TotalVendas;

--INDEX

CREATE INDEX index1 ON Produto (IdProduto);