ETEC-ZONA LESTE

DESENVOLVIMENTO DE SISTEMAS

João Pedro Silva de Oliveira

Atividade de Banco de Dados

Industria de beleza

São Paulo

2024

Criação do banco de dados:

```
CREATE DATABASE Indus_blz;
USE Indus_blz;
```

Tabelas:

```sql
-- Criando a tabela Regiao primeiro, pois ela é referenciada por outras tabelas
CREATE TABLE Regiao (
    Codigo INT PRIMARY KEY,
    Nome VARCHAR(100)
);

-- Criando a tabela Vendedor que referencia Regiao
CREATE TABLE Vendedor (
    Codigo INT PRIMARY KEY,
    Nome VARCHAR(100),
    RegiaoCodigo INT,
    FOREIGN KEY (RegiaoCodigo) REFERENCES Regiao(Codigo)
);

-- Criando a tabela Cliente
CREATE TABLE Cliente (
    Cod_Cliente INT PRIMARY KEY,
    Nome_Cliente VARCHAR(100),
    CPF VARCHAR(11) UNIQUE,
    Data_Nasc DATE,
    Endereco VARCHAR(200)
);

-- Criando a tabela NotaFiscal que faz referência a Vendedor e Cliente
CREATE TABLE NotaFiscal (
    Codigo INT PRIMARY KEY,
    Data DATE,
    VendedorCodigo INT,
    ClienteCodigo INT,
    FOREIGN KEY (VendedorCodigo) REFERENCES Vendedor(Codigo),
    FOREIGN KEY (ClienteCodigo) REFERENCES Cliente(Cod_Cliente)
);

-- Criando a tabela Produto
CREATE TABLE Produto (
    Codigo INT PRIMARY KEY,
    Nome VARCHAR(100),
    StatusVenda BIT
);
```

```sql
-- Criando a tabela NotaFiscalProduto que faz referência a NotaFiscal e Produto
CREATE TABLE NotaFiscalProduto (
    NotaFiscalCodigo INT,
    ProdutoCodigo INT,
    Quantidade INT,
    PRIMARY KEY (NotaFiscalCodigo, ProdutoCodigo),
    FOREIGN KEY (NotaFiscalCodigo) REFERENCES NotaFiscal(Codigo),
    FOREIGN KEY (ProdutoCodigo) REFERENCES Produto(Codigo)
);

-- Criando a tabela PontoEstrategico
CREATE TABLE PontoEstrategico (
    Codigo INT PRIMARY KEY,
    Nome VARCHAR(100),
    RegiaoCodigo INT,
    FOREIGN KEY (RegiaoCodigo) REFERENCES Regiao(Codigo)
);

-- Criando a tabela Veiculo que referencia Vendedor
CREATE TABLE Veiculo (
    Codigo INT PRIMARY KEY,
    Nome VARCHAR(100),
    VendedorCodigo INT,
    DataUso DATE,
    FOREIGN KEY (VendedorCodigo) REFERENCES Vendedor(Codigo)
);

-- Criando a tabela Escala que referencia Veiculo e Vendedor
CREATE TABLE Escala (
    Cod_Veiculo INT,
    Cod_Vendedor INT,
    Data DATE,
    PRIMARY KEY (Cod_Veiculo, Cod_Vendedor, Data),
    FOREIGN KEY (Cod_Veiculo) REFERENCES Veiculo(Codigo),
    FOREIGN KEY (Cod_Vendedor) REFERENCES Vendedor(Codigo)
);
```

Inserções:

```sql
-- Inserindo na tabela Veiculo
INSERT INTO Veiculo (Codigo, Nome, VendedorCodigo, DataUso)
VALUES
    (1, 'Gol GTI 2010', 1, '2024-10-10'),
    (2, 'Polo 2020', 2, '2024-11-02'),
    (3, 'HB 20 2015', 3, '2024-10-03');

-- Inserindo na tabela Escala
INSERT INTO Escala (Cod_Veiculo, Cod_Vendedor, Data)
VALUES
    (1, 1, '2024-09-15'),
    (2, 2, '2024-10-12'),
    (3, 3, '2024-11-03');

-- Inserindo na tabela NotaFiscal
INSERT INTO NotaFiscal (Codigo, Data, VendedorCodigo, ClienteCodigo)
VALUES
    (1, '2024-08-10', 1, 1),
    (2, '2024-06-11', 2, 2),
    (3, '2024-09-10', 3, 3),
    (4, '2024-11-15', 4, 4);

-- Inserindo na tabela NotaFiscalProduto
INSERT INTO NotaFiscalProduto (NotaFiscalCodigo, ProdutoCodigo, Quantidade)
VALUES
    (1, 1, 2),
    (1, 2, 1),
    (2, 3, 5),
    (3, 1, 3),
    (3, 2, 2);
```

```sql
-- Inserindo na tabela Regiao
INSERT INTO Regiao (Codigo, Nome)
VALUES (1, 'Norte'), (2, 'Sul'), (3, 'Leste'), (4, 'Oeste');

-- Inserindo na tabela PontoEstrategico
INSERT INTO PontoEstrategico (Codigo, Nome, RegiaoCodigo)
VALUES
    (1, 'Loja Central', 1),
    (2, 'Loja Sul', 2),
    (3, 'Loja Leste', 3),
    (4, 'Loja Oeste', 4);

-- Inserindo na tabela Vendedor
INSERT INTO Vendedor (Codigo, Nome, RegiaoCodigo)
VALUES
    (1, 'Ricardo Santos', 1),
    (2, 'Carla Machado', 2),
    (3, 'Lucas Ramos', 3),
    (4, 'Amanda Lima', 4);

-- Inserindo na tabela Cliente
INSERT INTO Cliente (Cod_Cliente, Nome_Cliente, CPF, Data_Nasc, Endereco)
VALUES
    (1, 'José Conceição', '12345678901', '1996-08-13', 'Rua Trevo de Santa Maria, 110'),
    (2, 'Flavia Nunes', '23456789012', '1998-10-25', 'Rua Fava, 58'),
    (3, 'Cristian Gomes', '34567890123', '2001-06-22', 'Rua Jaramataia, 96'),
    (4, 'Raissa Lopez', '45678901234', '1998-11-02', 'Rua Floco de Neve, 102');

-- Inserindo na tabela Produto
    INSERT INTO Produto (Codigo, Nome, StatusVenda)
    SELECT * FROM (VALUES
        (1, 'Shampoo', 1),
        (2, 'Condicionador', 1),
        (3, 'Sabonete', 1),
        (4, 'Creme Capilar', 0),
        (5, 'Perfume', 1),
        (6, 'Desodorante', 0),
        (7, 'Batom', 1),
        (8, 'Gloss', 1),
        (9, 'Gilete', 1)
    ) AS NewProducts (Codigo, Nome, StatusVenda)
    WHERE NOT EXISTS (
        SELECT 1 FROM Produto WHERE Produto.Codigo = NewProducts.Codigo
    );
```

Comandos:

Comando1:

```sql
-- Consulta para listar os pontos estratégicos de cada região.
SELECT Regiao.Nome AS Regiao, PontoEstrategico.Nome AS PontoEstrategico
FROM Regiao
JOIN PontoEstrategico ON Regiao.Codigo = PontoEstrategico.RegiaoCodigo;
```

83 %

Resultados | Mensagens

	Regiao	PontoEstrategico
1	Norte	Loja Central
2	Sul	Loja Sul
3	Leste	Loja Leste
4	Oeste	Loja Oeste

Comando2:

```sql
-- Consulta para listar todas as regiões cadastradas.
SELECT Nome FROM Regiao;
```

83 %

Resultados | Mensagens

	Nome
1	Norte
2	Sul
3	Leste
4	Oeste

Comando3:

```sql
-- Consulta para listar os vendedores e quais veículos utilizaram no último mês.
SELECT Vendedor.Nome AS Nome_Vendedor, Veiculo.Codigo AS Codigo_Veiculo, Veiculo.Nome AS Nome_Veiculo
FROM Vendedor
JOIN Veiculo ON Vendedor.Codigo = Veiculo.VendedorCodigo
WHERE Veiculo.DataUso BETWEEN DATEADD(MONTH, -1, GETDATE()) AND GETDATE();
```

83 %

Resultados | Mensagens

	Nome_Vendedor	Codigo_Veiculo	Nome_Veiculo
1	Ricardo Santos	1	Gol GTI 2010
2	Lucas Ramos	3	HB 20 2015

Comando4:

```sql
-- Consulta para listar os vendedores e suas respectivas regiões.
SELECT Vendedor.Nome AS Nome_Vendedor, Regiao.Nome AS Nome_Regiao
FROM Vendedor
JOIN Regiao ON Vendedor.RegiaoCodigo = Regiao.Codigo;
```

83 %

Resultados | Mensagens

	Nome_Vendedor	Nome_Regiao
1	Ricardo Santos	Norte
2	Carla Machado	Sul
3	Lucas Ramos	Leste
4	Amanda Lima	Oeste

Comando5:

```sql
-- Consulta para obter os produtos de um vendedor específico
SELECT
    Produto.Codigo AS Codigo_Produto,
    Produto.Nome AS Nome_Produto,
    Produto.StatusVenda,
    NotaFiscal.Codigo AS Codigo_NotaFiscal,
    NotaFiscal.Data AS Data_NotaFiscal,
    NotaFiscalProduto.Quantidade AS Quantidade_Produto_Vendido
FROM
    NotaFiscal
JOIN
    NotaFiscalProduto ON NotaFiscal.Codigo = NotaFiscalProduto.NotaFiscalCodigo
JOIN
    Produto ON NotaFiscalProduto.ProdutoCodigo = Produto.Codigo
WHERE
    NotaFiscal.VendedorCodigo = 1;
```

83 %

Resultados | Mensagens

	Codigo_Produto	Nome_Produto	StatusVenda	Codigo_NotaFiscal	Data_NotaFiscal	Quantidade_Produto_Vendido
1	1	Shampoo	1	1	2024-08-10	2
2	2	Condicionador	1	1	2024-08-10	1

Comando6:

```sql
-- Consulta para listar os vendedores que venderam um determinado produto
SELECT Vendedor.Nome AS Nome_Vendedor
FROM Vendedor
JOIN NotaFiscal ON Vendedor.Codigo = NotaFiscal.VendedorCodigo
JOIN NotaFiscalProduto ON NotaFiscal.Codigo = NotaFiscalProduto.NotaFiscalCodigo
WHERE NotaFiscalProduto.ProdutoCodigo = 1;
```

83 %

Resultados | Mensagens

	Nome_Vendedor
1	Ricardo Santos
2	Lucas Ramos

Comando7:

```sql
-- Consulta para listar os produtos que ainda não foram vendidos.
SELECT Nome FROM Produto WHERE StatusVenda = 0;
```

83 %

Resultados | Mensagens

	Nome
1	Creme Capilar
2	Desodorante

Comando8:

```sql
-- Consulta para listar os detalhes de um veículo específico
SELECT * FROM Veiculo WHERE Codigo = 1;
```

83 %

Resultados | Mensagens

	Codigo	Nome	VendedorCodigo	DataUso
1	1	Gol GTI 2010	1	2024-10-10

Comando9:

```sql
-- Consulta para obter a quantidade de itens vendidos em cada nota fiscal.
SELECT NotaFiscal.Codigo AS Codigo_NotaFiscal, SUM(NotaFiscalProduto.Quantidade) AS Total_Itens
FROM NotaFiscal
JOIN NotaFiscalProduto ON NotaFiscal.Codigo = NotaFiscalProduto.NotaFiscalCodigo
GROUP BY NotaFiscal.Codigo;
```

83 %

Resultados | Mensagens

	Codigo_NotaFiscal	Total_Itens
1	1	3
2	2	5
3	3	5