



Campus: Estácio de Sá Porto Velho Polo São Cristovão

Curso: Desenvolvimento Full Stack

Disciplina: Nível 2: Vamos Manter as Informações?

Semestre: 2024.4 Full Stack

Integrantes: Ismael Lucena de Albuquerque

Git: <https://github.com/ismaellucena/Miss-o-Pr-tica-N-vel-2-Mundo-3.git>

Relatório - Missão Prática | Nível 2 | Mundo 3

Objetivo da Prática

Este projeto tem como objetivo a modelagem e implementação de um banco de dados simples,

utilizando o SQL Server. O foco é permitir que os alunos vivenciem a experiência prática de modelar, criar e manipular dados em um banco relacional.

Script de Criação do Banco de Dados e Tabelas

```
CREATE DATABASE LojaDB;
```

```
GO
```

```
USE LojaDB;
```

```
GO
```

```
CREATE TABLE Usuarios (
```

```
    UsuarioID INT IDENTITY(1,1) PRIMARY KEY,
```

```
    Nome NVARCHAR(100),
```

```
    Senha NVARCHAR(50)
```

```
);
```

```
CREATE TABLE Pessoas (
```

```
    PessoaID INT PRIMARY KEY,
```

```
    Nome NVARCHAR(100),
```

```
    Endereco NVARCHAR(200),
```

```
    Telefone NVARCHAR(15),
```

```
    TipoPessoa CHAR(1) CHECK (TipoPessoa IN ('F', 'J'))
```

```
);
```

```
CREATE TABLE PessoaFisica (
```

```
PessoaID INT PRIMARY KEY FOREIGN KEY REFERENCES  
Pessoas(PessoaID),
```

```
CPF CHAR(11) UNIQUE
```

```
);
```

```
CREATE TABLE PessoaJuridica (
```

```
PessoaID INT PRIMARY KEY FOREIGN KEY REFERENCES  
Pessoas(PessoaID),
```

```
CNPJ CHAR(14) UNIQUE
```

```
);
```

```
CREATE TABLE Produtos (
```

```
ProdutoID INT IDENTITY(1,1) PRIMARY KEY,
```

```
Nome NVARCHAR(100),
```

```
Quantidade INT,
```

```
PrecoVenda DECIMAL(10, 2)
```

```
);
```

```
CREATE TABLE Movimentacoes (
```

```
MovimentacaoID INT IDENTITY(1,1) PRIMARY KEY,
```

```
TipoMovimentacao CHAR(1) CHECK (TipoMovimentacao IN ('E', 'S')),
```

```
ProdutoID INT FOREIGN KEY REFERENCES Produtos(ProdutoID),  
PessoaID INT FOREIGN KEY REFERENCES Pessoas(PessoaID),  
Quantidade INT,  
PrecoUnitario DECIMAL(10, 2),  
ValorTotal AS (Quantidade * PrecoUnitario)  
);
```

Script de Inserção de Dados

```
INSERT INTO Usuarios (Nome, Senha)  
VALUES ('op1', 'op1'), ('op2', 'op2');
```

```
INSERT INTO Pessoas (PessoaID, Nome, Endereco, Telefone, TipoPessoa)  
VALUES (1, 'João Silva', 'Rua A, 123', '11999999999', 'F'),  
      (2, 'Empresa X', 'Rua B, 456', '11222222222', 'J');
```

```
INSERT INTO PessoaFisica (PessoaID, CPF)  
VALUES (1, '12345678901');
```

```
INSERT INTO PessoaJuridica (PessoaID, CNPJ)  
VALUES (2, '12345678000101');
```

```
INSERT INTO Produtos (Nome, Quantidade, PrecoVenda)  
VALUES ('Produto A', 50, 10.00), ('Produto B', 30, 20.00);
```

```
INSERT INTO Movimentacoes (TipoMovimentacao, ProdutoID, PessoaID,  
Quantidade,  
PrecoUnitario)
```

```
VALUES ('E', 1, 2, 10, 8.00),  
      ('S', 1, 1, 5, 10.00);
```

Consultas Realizadas

-- Listar todas as pessoas físicas

```
SELECT * FROM Pessoas P  
JOIN PessoaFisica PF ON P.PessoaID = PF.PessoaID;
```

-- Listar todas as pessoas jurídicas

```
SELECT * FROM Pessoas P  
JOIN PessoaJuridica PJ ON P.PessoaID = PJ.PessoaID;
```

-- Movimentações de entrada

```
SELECT * FROM Movimentacoes M  
JOIN Produtos P ON M.ProdutoID = P.ProdutoID  
WHERE TipoMovimentacao = 'E';
```

-- Valor total das saídas agrupado por produto

```
SELECT P.Nome AS Produto, SUM(M.ValorTotal) AS TotalSaida  
FROM Movimentacoes M  
JOIN Produtos P ON M.ProdutoID = P.ProdutoID  
WHERE TipoMovimentacao = 'S'
```