Endereço do Campus/Polo:

Rua Morais E Silva 40 - Maracanã - Rio de Janeiro - RJ - CEP: 20.271-030

Curso: Desenvolvimento Full Stack

Disciplina: Vamos manter as informações?

Turma: RPG0015 / 9001

Semestre Letivo: Terceiro

Nome: Matheus Zimmer Moreira Martins

Endereço do Repositório:

https://github.com/Mazimo-Marts/Trabalho-Mundo3-Nvl2

Alimentando a Base

Objetivo:

Identificar os requisitos de um sistema e transformá-los no modelo adequado. Utilizar ferramentas de modelagem para bases de dados relacionais. Explorar a sintaxe SQL na criação das estruturas do banco (DDL). Explorar a sintaxe SQL na consulta e manipulação de dados (DML) No final do exercício, o aluno terá vivenciado a experiência de modelar a base de dados para um sistema simples, além de implementá-la, através da sintaxe SQL, na plataforma do SQL Server.

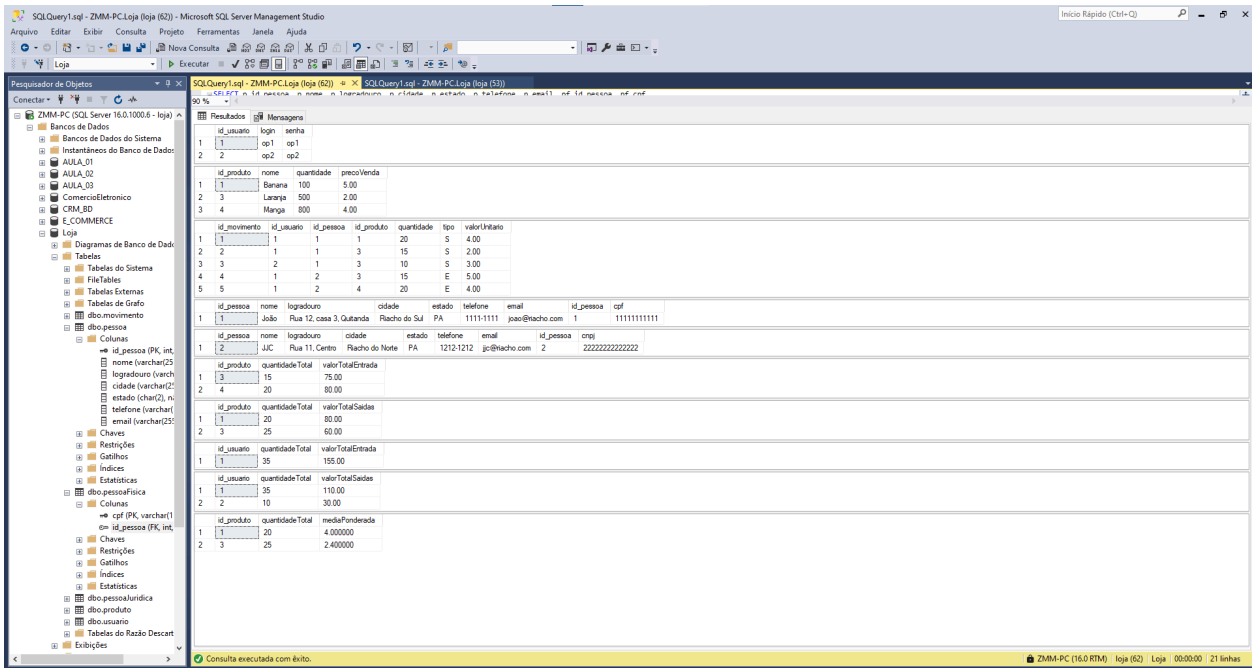
Códigos da Prática:

SQLQuery1.sql - ZMM-PC.Loja (loja (62))*

```sql
-- Criando tabela Usuários
CREATE TABLE usuario(
    id_usuario INT IDENTITY(1,1) PRIMARY KEY,
    login VARCHAR(50) NOT NULL UNIQUE,
    senha VARCHAR(50) NOT NULL
);
GO

-- Criando tabela Produtos
CREATE TABLE produto(
    id_produto INT NOT NULL PRIMARY KEY,
    nome VARCHAR(25) NOT NULL,
    quantidade INT NOT NULL,
    precoVenda DECIMAL(10,2) NOT NULL
);
GO


-- Criando tabela Movimentação
CREATE TABLE movimento(
    id_movimento INT IDENTITY(1,1) PRIMARY KEY,
    id_usuario INT NOT NULL,
    id_pessoa INT NOT NULL,
    id_produto INT NOT NULL,
    quantidade INT NOT NULL,
    tipo CHAR(1) NOT NULL CHECK (tipo IN ('E', 'S')),
    valorUnitario DECIMAL(10,2) NOT NULL,
    FOREIGN KEY(id_usuario) REFERENCES usuario(id_usuario),
    FOREIGN KEY(id_pessoa) REFERENCES pessoa(id_pessoa),
    FOREIGN KEY(id_produto) REFERENCES produto(id_produto)
);
GO

-- Inserindo usuários na DB
INSERT INTO usuario VALUES
    ('op1', 'op1'),
    ('op2', 'op2');
GO

-- Inserindo produtos na DB
INSERT INTO produto VALUES
    (1, 'Banana', 100, 5.00),
    (3, 'Laranja', 500, 2.00),
    (4, 'Manga', 800, 4.00);
GO

-- Inserindo pessoas na DB
INSERT INTO pessoa VALUES
    ('João', 'Rua 12, casa 3, Quitanda', 'Riacho do Sul', 'PA', '1111-1111', 'joao@riacho.com'),
    ('JJC', 'Rua 11, Centro', 'Riacho do Norte', 'PA', '1212-1212', 'jjc@riacho.com');
GO
```

```sql
-- Inserindo pessoasFisicas na DB
INSERT INTO pessoaFisica VALUES ('11111111111', 1);
GO

-- Inserindo pessoasJuridicas na DB
INSERT INTO pessoaJuridica VALUES ('22222222222222', 2);
GO

-- Inserindo movimentações na DB
INSERT INTO movimento VALUES
    (1, 1, 1, 20, 'S', 4.00),
    (1, 1, 3, 15, 'S', 2.00),
    (2, 1, 3, 10, 'S', 3.00),
    (1, 2, 3, 15, 'E', 5.00),
    (1, 2, 4, 20, 'E', 4.00);
GO
```

Resultado da Execução:



Análise:

a) Quais as diferenças no uso de sequence e identity?

R:

IDENTITY é uma propriedade de coluna vinculada a uma tabela específica, que gera valores automaticamente durante inserções, com configurações limitadas. SEQUENCE, por outro lado, é um objeto independente no banco de dados que pode ser compartilhado entre várias tabelas, oferece maior controle sobre a geração de valores (incluindo obtenção antecipada e opção de ciclo), e possui mais opções de personalização.

b) Qual a importância das chaves estrangeiras para a consistência do banco?

R:

Elas garantem a integridade referencial ao impedir referências a dados inexistentes, e permitem definir ações automáticas quando registros relacionados são modificados, e também estabelecem e mantêm relações válidas entre tabelas.

c) Quais operadores do SQL pertencem à álgebra relacional e quais são definidos no cálculo relacional?

R:

Álgebra Relacional: SELECT / FROM / WHERE / Operações de conjunto / GROUP BY / DINSTINCT

Cálculo Relacional: EXISTS / IN / NOT IN/ ALL/ ANY / HAVING

d) Como é feito o agrupamento em consultas, e qual requisito é obrigatório?

R:

O agrupamento é feito usando a cláusula `GROUP BY`, que organiza os registros em grupos com base nos valores de uma ou mais colunas especificadas. O requisito obrigatório ao usar agrupamento é que toda coluna incluída na lista SELECT que não esteja dentro de uma função agregada deve aparecer na cláusula GROUP BY.