

Disciplina: Nível 2: Vamos Manter as Informações

Semestre: 2º - 2024

Aluno: GILBERTO FERREIRA DA SILVA JUNIOR

Matrícula: 2023.0701.4923

Repositório do Projeto:

(RPG0015---Vamos Manter as Informações (github.com))

Relatório discente de acompanhamento

Título da Prática: RPG0015 – Vamos Manter as Informações

Objetivo da Prática:

O objetivo principal desta atividade prática é aplicar os conceitos teóricos de modelagem de banco de dados, por meio das ferramentas Microsoft SQL Server como plataforma e do DB Designer. Através da criação de um banco de dados para o projeto fictício de uma loja, o objetivo é consolidar o entendimento sobre a estruturação dos dados em ambientes relacionais.

De forma mais específica os objetivos principais que norteiam a prática são:

1 – A criação do banco de dados denominado “Loja” composto pelas tabelas para armazenar informações sobre pessoas, usuários, produtos e movimentações inerentes ao negócio.

2 – Municar as tabelas criadas com dados fictícios, abrangendo aspectos tais como: pessoas físicas e jurídicas, usuários, produtos e movimentações de entradas e saídas.

Estes objetivos específicos visam não apenas a aplicação prática dos conceitos, mas também uma análise crítica dos resultados, permitindo uma avaliação completa do processo de modelagem e implementação de banco de dados.

Metodologia:

Para a construção e implementação do banco de dados, considerando a utilização das ferramentas DB Designer para a modelagem visual do projeto.

Inicialmente definimos a estrutura do banco de dados por meio de um modelo visual, para criar esquemas conceituais, identificando entidades, atributos e as relações entre eles. Desta forma foi possível abordar de forma intuitiva na elaboração do projeto.



Vale ressaltar que a ferramenta DB Designer oferece a possibilidade de exportar automaticamente os esboços visuais gerando desta forma os códigos SQL necessários para o banco de dados.

Concluída esta fase do projeto, os códigos gerados no DB Designer foram utilizados no SQL Server Management Studio, garantindo fidelidade entre o modelo conceitual e a implementação prática.

O ambiente SQL Server Management Studio foi a ferramenta utilizada para a construção efetiva do banco de dados no servidor SQL Server. Os códigos foram aplicados e adaptados conforme necessário, pois é preciso considerar as características específicas do SQL Server.

Para a simulação e testes foram implementados dados fictícios nas tabelas do banco, por meio de Scripts SQL no SQL Server Management Studio, seguindo as diretrizes definidas em cada tabela.

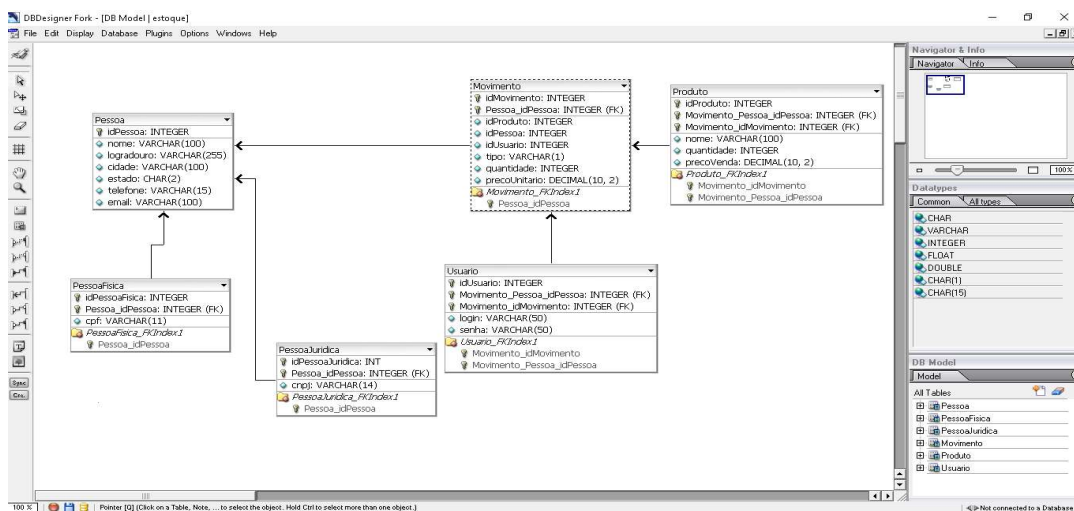
A fase inicial contou com a elaboração de consultas SQL para testar, extrair e analisar informações do banco de dados. Foram formuladas consultas que abordaram dados específicos em todas as tabelas.

Desta forma, a combinação das ferramentas DB Designer e SQL Server Management Studio mostrou-se uma estratégia eficaz para o desenvolvimento da atividade prática, permitindo uma transição leve para a implementação do projeto no ambiente de servidor.

Resultados

Com isso, os resultados da prática de modelagem e implementação de banco de dados foram:

A - Modelagem do banco de dados utilizando a Ferramenta DB Designer.





B – Criação da estrutura do banco de dados no SQL Server Management Studio.

```
USE master;
GO

-- Criar o login
CREATE LOGIN loja WITH PASSWORD = 'loja', CHECK_POLICY = OFF;
GO

-- Criar o banco de dados
CREATE DATABASE loja;
GO

-- Usar o banco de dados
USE loja;
GO

-- Criar o usuário associado ao login
CREATE USER loja FOR LOGIN loja;
GO

-- Criar as tabelas (como já feito anteriormente)
CREATE TABLE Usuarios (
    id_usuario INT IDENTITY(1,1) PRIMARY KEY,
    nome_usuario VARCHAR(100),
    senha_usuario VARCHAR(100)
);

CREATE TABLE Pessoas (
    id_pessoa INT IDENTITY(1,1) PRIMARY KEY,
    nome VARCHAR(100),
    endereco VARCHAR(255),
    telefone VARCHAR(20),
    tipo_pessoa CHAR(1) -- 'F' para Física, 'J' para Jurídica
);

CREATE TABLE PessoaFisica (
    cpf VARCHAR(11) PRIMARY KEY,
    id_pessoa INT FOREIGN KEY REFERENCES Pessoas(id_pessoa) ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE PessoaJuridica (
    cnpj VARCHAR(14) PRIMARY KEY,
    id_pessoa INT FOREIGN KEY REFERENCES Pessoas(id_pessoa) ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Produtos (
    id_produto INT IDENTITY(1,1) PRIMARY KEY,
    nome_produto VARCHAR(100),
    quantidade INT,
    preco_venda DECIMAL(10, 2)
);

CREATE TABLE Movimentacoes (
    id_movimento INT IDENTITY(1,1) PRIMARY KEY,
    tipo_movimento CHAR(1) CHECK (tipo_movimento IN ('E', 'S')), -- 'E' para Entrada e 'S' para Saída
    id_produto INT FOREIGN KEY REFERENCES Produtos(id_produto) ON DELETE CASCADE ON UPDATE CASCADE,
    id_pessoa INT FOREIGN KEY REFERENCES Pessoas(id_pessoa) ON DELETE CASCADE ON UPDATE CASCADE,
    quantidade INT,
    preco_unitario DECIMAL(10, 2)
);
```

50 %

Conectado. (1/1)

C – Alimentação do banco de dados com dados fictícios.

```
USE loja;

INSERT INTO Usuarios (nome_usuario, senha_usuario)
VALUES ('op1', 'op1'), ('op2', 'op2');

INSERT INTO Produtos (nome_produto, quantidade, preco_venda)
VALUES
('Produto A', 10, 20.00),
('Produto B', 5, 30.00),
('Produto C', 20, 15.00);

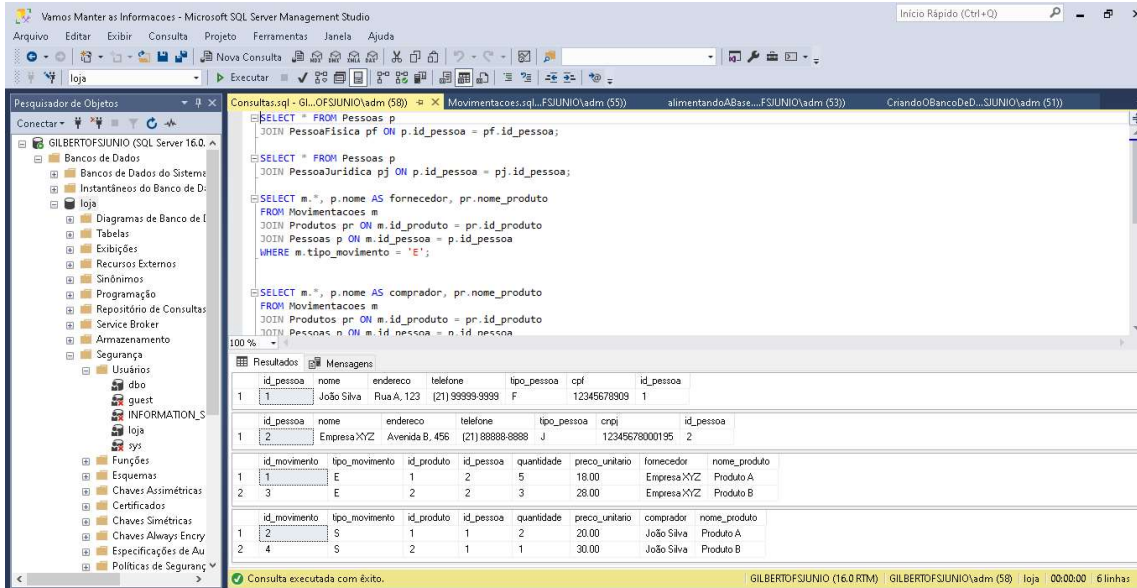
INSERT INTO Pessoas (nome, endereco, telefone, tipo_pessoa)
VALUES
('João Silva', 'Rua A, 123', '(21) 99999-9999', 'F'), -- F para Física
('Empresa XYZ', 'Avenida B, 456', '(21) 88888-8888', 'J'); -- J para Jurídica

INSERT INTO PessoaFisica (cpf, id_pessoa)
VALUES ('12345678909', (SELECT id_pessoa FROM Pessoas WHERE nome = 'João Silva'));

INSERT INTO PessoaJuridica (cnpj, id_pessoa)
VALUES ('12345678900195', (SELECT id_pessoa FROM Pessoas WHERE nome = 'Empresa XYZ'));
```

100 %

Conectado. (1/1)

D – execução de consultas SQL

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The left pane shows the server hierarchy for 'GILBERTOFSIJUNIO (SQL Server 16.0)'. The central pane contains four SQL queries. The bottom pane shows the results of the last query, which is a complex join involving 'Movimentos', 'Produtos', and 'Pessoas' tables.

Query 1:

```
SELECT * FROM Pessoas p
JOIN PessoaFisica pf ON p.id_pessoa = pf.id_pessoa;
```

Query 2:

```
SELECT * FROM Pessoas p
JOIN PessoaJuridica pj ON p.id_pessoa = pj.id_pessoa;
```

Query 3:

```
SELECT m.*, p.nome AS fornecedor, pr.nome_produto
FROM Movimentos m
JOIN Produtos pr ON m.id_produto = pr.id_produto
JOIN Pessoas p ON m.id_pessoa = p.id_pessoa
WHERE m.tipo_movimento = 'E';
```

Query 4:

```
SELECT m.*, p.nome AS comprador, pr.nome_produto
FROM Movimentos m
JOIN Produtos pr ON m.id_produto = pr.id_produto
JOIN Pessoas p ON m.id_pessoa = p.id_pessoa;
```

Results of Query 4:

id_pessoa	nome	endereco	telefone	tipo_pessoa	cpf	id_pessoa
1	João Silva	Rua A, 123	(21) 99999-9999	F	12345678909	1
2	Empresa XYZ	Avenida B, 456	(21) 88888-8888	J	12345678000195	2

id_movimento	tipo_movimento	id_produto	id_pessoa	quantidade	preco_unitario	fornecedor	nome_produto
1	E	1	2	5	18.00	Empresa XYZ	Produto A
2	E	2	2	3	28.00	Empresa XYZ	Produto B

id_movimento	tipo_movimento	id_produto	id_pessoa	quantidade	preco_unitario	comprador	nome_produto
1	S	1	2	2	20.00	João Silva	Produto A
2	S	2	1	1	30.00	João Silva	Produto B

Consulta executada com êxito. GILBERTOFSIJUNIO (16.0 RTM) GILBERTOFSIJUNIO\adm (58) loja 00:00:00 6 linhas