



Curso: Desenvolvimento Full Stack 2024.1 3º Semestre

Aluno: Rogério Benedito Geraldo

Polo: Nova Ourinhos

Tutor: Alessandro dos Santos

Disciplina: Vamos desenvolver as informações

Objetivos da prática:

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).

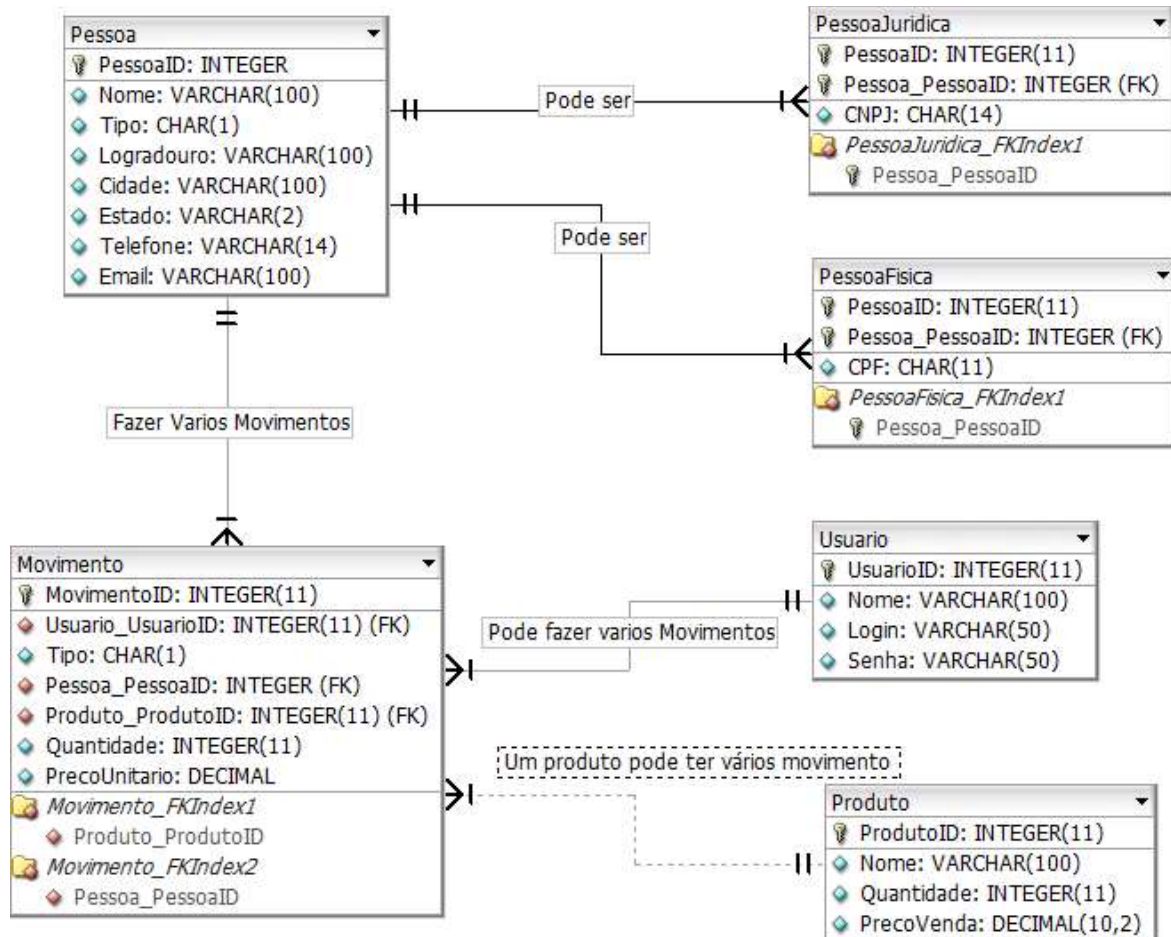
Explorar a sintaxe SQL na consulta e manipulação de dados (DML)

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.

Link do Git: <https://github.com/Rogeriobg/vamos-Manter-As-Informacoes.git>

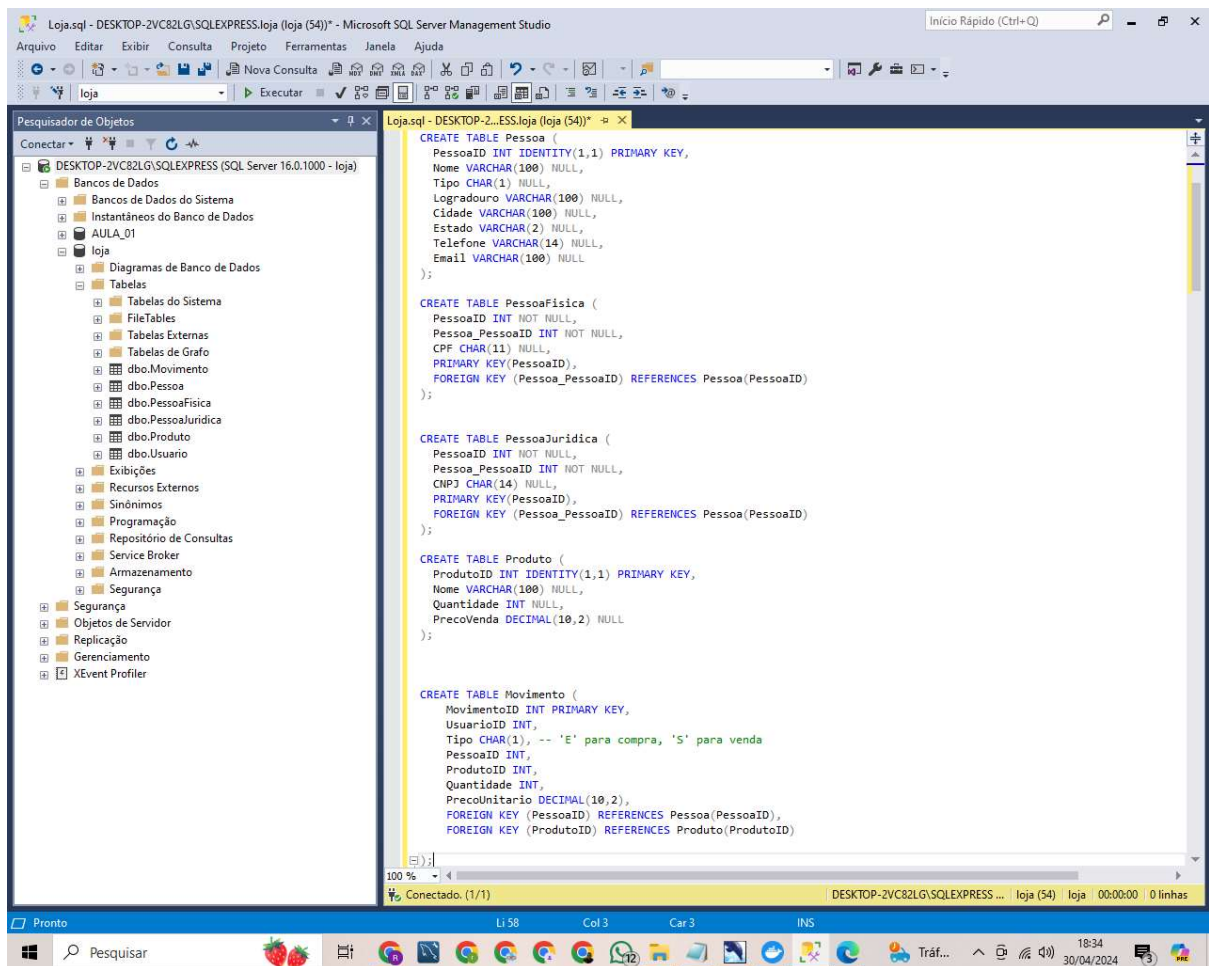
Códigos do 1º Procedimentos

Modelagem feita no DBDesigner 4





Print do Projeto:





Código do script SQL utilizando SQL Server Management Studio:

```
CREATE DATABASE Loja;
```

```
GO
```

```
USE Loja;
```

```
GO
```

```
USE loja;
```

```
CREATE TABLE Usuario (
```

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

```
    Login VARCHAR(50) NULL,
```

```
    Senha VARCHAR(50) NULL
```

```
);
```

```
CREATE TABLE Pessoa (
```

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

```
    Nome VARCHAR(100) NULL,
```



Tipo CHAR(1) NULL,
Logradouro VARCHAR(100) NULL,
Cidade VARCHAR(100) NULL,
Estado VARCHAR(2) NULL,
Telefone VARCHAR(14) NULL,
Email VARCHAR(100) NULL
);

```
CREATE TABLE PessoaFisica (  
    PessoaID INT NOT NULL,  
    Pessoa_PessoaID INT NOT NULL,  
    CPF CHAR(11) NULL,  
    PRIMARY KEY(PessoaID),  
    FOREIGN KEY (Pessoa_PessoaID) REFERENCES Pessoa(PessoaID)  
);
```

```
CREATE TABLE PessoaJuridica (  
    PessoaID INT NOT NULL,  
    Pessoa_PessoaID INT NOT NULL,  
    CNPJ CHAR(14) NULL,  
    PRIMARY KEY(PessoaID),  
    FOREIGN KEY (Pessoa_PessoaID) REFERENCES Pessoa(PessoaID)  
);
```



```
CREATE TABLE Produto (  
    ProdutoID INT IDENTITY(1,1) PRIMARY KEY,  
    Nome VARCHAR(100) NULL,  
    Quantidade INT NULL,  
    PreçoVenda DECIMAL(10,2) NULL  
);
```

```
CREATE TABLE Movimento (  
    MovimentoID INT PRIMARY KEY,  
    UsuarioID INT,  
    Tipo CHAR(1), -- 'E' para compra, 'S' para venda  
    PessoaID INT,  
    ProdutoID INT,  
    Quantidade INT,  
    PreçoUnitario DECIMAL(10,2),  
    FOREIGN KEY (PessoaID) REFERENCES Pessoa(PessoaID),  
    FOREIGN KEY (ProdutoID) REFERENCES Produto(ProdutoID)  
);
```

Conclusão e Análise:

Como são implementadas as diferentes cardinalidades, basicamente 1X1, 1XN ou NxN, em um banco de dados relacional?



1X1: Chave estrangeira única em uma das tabelas referenciando a chave primária da outra tabela.

1XN: Chave estrangeira na tabela "N" referenciando a chave primária da tabela "1". NxN: Tabela de associação contendo chaves estrangeiras referenciando as tabelas relacionadas.

Que tipo de relacionamento deve ser utilizado para representar o uso de herança em bancos de dados relacionais?

Relacionamento de tabela única com chaves estrangeiras opcionais.

Como o SQL Server Management Studio permite a melhoria da produtividade nas tarefas relacionadas ao gerenciamento do banco de dados?

Oferecendo uma interface intuitiva para consultas, administração e visualização de objetos do banco de dados.