



**Nome do Campus:** Polo Cruzeiro-Sp

**Nome do Curso:** Desenvolvimento FullStack

**Nome da Disciplina:** Vamos Manter as informações?

**Número da Turma:** 2025.1

**Semestre Letivo:** Primeiro Semestre

**Nome do Aluno:** Moniza de Oliveira Silva Santos Pelegrini

**Matrícula:** 202401190829

**Título da Prática:** Modelagem e implementação de um banco de dados simples, utilizando como base o SQL SERVER.

**Objetivo da Prática:** Identificar os requisitos de um sistema e transformá-los no modelo adequado;

Utilizar ferramentas de modelagem para a base de dados relacionais;

Explorar a sintaxe SQL na criação das estruturas do banco (DDL);

Explorar a sintaxe SQL na criação das estruturas do banco (DML);

Utilizar persistência de objetos em arquivos binários:

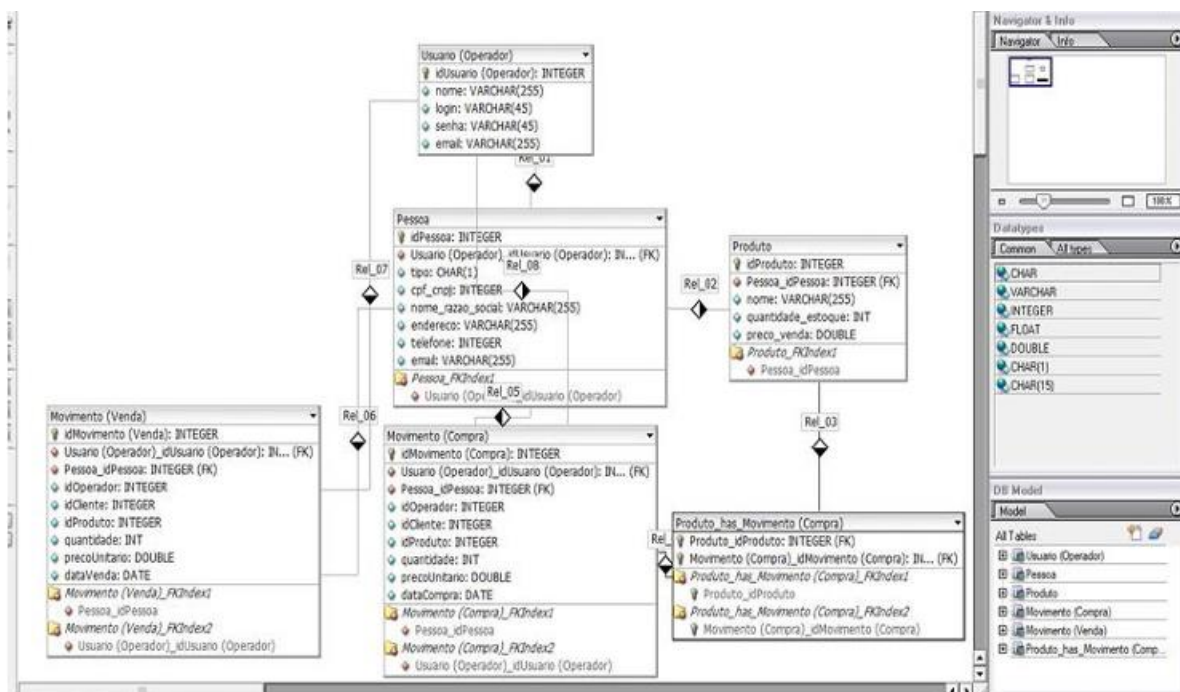
Implementar métodos para salvar e recuperar dados em arquivos binários usando serialização.

Utilizar persistência de objetos em arquivos binários:

Implementar métodos para salvar e recuperar dados em arquivos binários usando serialização.

**Primeiro Procedimento:** Criando o Banco de Dados Códigos utilizados neste roteiro:

Códigos utilizados neste roteiro:



```
-- Criação da sequence para identificadores de pessoa
CREATE SEQUENCE seq_pessoa_id
  START WITH 1
  INCREMENT BY 1
  MINVALUE 1
  NO MAXVALUE
  CACHE 10;
GO
```

## Análise e Conclusão

1. Como são implementadas as diferentes cardinalidades (1x1, 1xN, NxN) em um banco de dados relacional?

Em bancos de dados relacionais, as cardinalidades representam como as entidades (tabelas) se relacionam entre si. As mais comuns são: o 1x1 (um para um): implementada com uma chave primária (PK) em uma tabela e uma chave estrangeira (FK) na outra, podendo ser única para garantir a relação exclusiva. o 1xN (um para muitos): uma chave primária em uma tabela se relaciona com várias ocorrências em outra, usando uma FK na tabela do "muitos". o NxN (muitos para muitos): exige a criação de uma tabela intermediária (tabela de junção) contendo FKs apontando para as duas tabelas principais.

2. Qual tipo de relacionamento é utilizado para representar herança em bancos de dados relacionais?

Embora bancos relacionais não possuam suporte nativo para herança como em linguagens orientadas a objetos, é possível simular esse conceito por meio de estratégias como: o Tabela única para toda a hierarquia (Single Table Inheritance) o Uma tabela por subclasse (Class Table Inheritance) o Uma tabela por classe concreta (Concrete Table Inheritance) A escolha da abordagem depende do grau de reutilização de atributos, da complexidade das consultas e do desempenho desejado.

3. Como o SQL Server Management Studio (SSMS) contribui para a produtividade no gerenciamento de bancos de dados?

O SSMS é uma ferramenta completa desenvolvida pela Microsoft para administração do SQL Server. Entre suas principais contribuições para a produtividade estão: o Interface gráfica intuitiva para criação, modificação e exclusão de objetos de banco de dados o Recursos avançados de depuração e execução de scripts SQL o Ferramentas para análise de desempenho, backup e restauração de dados o Suporte a tarefas automatizadas e agendamento de jobs o Integração com outras soluções Microsoft para maior eficiência no desenvolvimento e manutenção de bancos