



# Estácio

Faculdade Estácio - Polo Centro - Canela - RS

Curso: Desenvolvimento Full Stack

Disciplina: RPG0015 - Vamos manter as informações?

Turma: 9001 - Semestre Letivo: 2025.1 - 3º semestre

Integrante: Pedro Henrique Marques Medeiros Pinho

Matrícula: 202402031831

IDE: Sql Server Management Studio

Repositório Git: <https://github.com/PedroPinho23/Vamos-Manter-as-Informa-es.git>

## Vamos manter as informações?

Modelagem e implementação de um banco de dados simples, utilizando como base o SQL Server.

### 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.

## Códigos Utilizados:

use loja

INSERT INTO Usuario (login, senha)

VALUES

('op1', 'op1'),

('op2', 'op2'),

('op3', 'op3'),

('op4', 'op4');

SELECT \* FROM Usuario;

INSERT INTO Produto (nome, quantidade, precoVenda)

VALUES

('banana', 100, 5.0),

('laranja', 500, 2.0),

('Manga', 800, 4.0),

('Pera', 300, 6.0),

('Abacaxi', 300, 3.0);

SELECT \* FROM Produto;

INSERT INTO Pessoa (nome, rua, cidade, estado, telefone, email)

VALUES

('Joao', 'Rua 12, casa3, Quitanda', 'Riacho do Sul', 'PA', '11111111',  
'joao@riacho.com'),

('JJC', 'Rua 11, Centro', 'Riacho do Norte', 'PA', '12121212', 'jjc@riacho.com');

SELECT \* FROM Pessoa;

INSERT INTO PessoaFisica (idPessoa, cpf)

VALUES

(1, '11111111111');

SELECT \* FROM PessoaFisica;

INSERT INTO PessoaJuridica (idPessoa, cnpj)

VALUES

(2, '222222222222');

```
SELECT * FROM PessoaJuridica;
```

```
INSERT INTO Movimento (idUserario, idPessoa, idProduto, quantidade, tipo,  
valorUnitario)
```

```
VALUES
```

```
    (1, 1, 1, 10, 'E', 3.0),  
    (1, 1, 2, 10, 'E', 1.0),  
    (1, 1, 3, 10, 'E', 3.0),  
    (2, 2, 1, 3, 'S', 5.0),  
    (2, 2, 2, 1, 'S', 2.0),  
    (2, 2, 3, 8, 'S', 4.0),  
    (3, 2, 1, 5, 'E', 4.75),  
    (3, 2, 2, 7, 'E', 1.32),  
    (3, 1, 1, 9, 'S', 5.0),  
    (3, 1, 2, 3, 'S', 2.0),  
    (4, 2, 1, 8, 'S', 5.0),  
    (4, 2, 2, 6, 'S', 2.0);
```

```
SELECT * FROM Movimento;
```

## Resultados:

100 %							
Resultados				Mensagens			
	idProduto	nome	quantidade	precoVenda			
1	1	banana	100	5			
2	2	laranja	500	2			
3	3	Manga	800	4			
4	4	Pera	300	6			
5	5	Abacaxi	300	3			
	idPessoa	nome	rua	cidade	estado	telefone	email
1	1	Joao	Rua 12, casa3, Quitanda	Riacho do Sul	PA	11111111	joao@riacho.com
2	2	JJC	Rua 11, Centro	Riacho do Norte	PA	12121212	jjo@riacho.com
	idPessoa	cpf					
1	1	11111111111					
	idPessoa	cnpj					
1	2	22222222222222					
	idMovimento	idUsuario	idPessoa	idProduto	quantidade	tipo	valorUnitario
1	1	1	1	1	10	E	3
2	2	1	1	2	10	E	1
3	3	1	1	3	10	E	3
4	4	2	2	1	3	S	5
5	5	2	2	2	1	S	2
6	6	2	2	3	8	S	4
7	7	3	2	1	5	E	4,75
8	8	3	2	2	7	E	1,32
9	9	3	1	1	9	S	5
10	10	3	1	2	3	S	2
11	11	4	2	1	8	S	5
12	12	4	2	2	6	S	2

# Conclusão:

## **Quais as diferenças no uso de sequence e identity?**

A diferença principal entre sequence e identity é que o identity é usado diretamente em uma coluna de uma tabela para gerar valores automáticos e sequenciais, enquanto a sequence é um objeto separado que pode ser usado em várias tabelas ou operações. O identity é mais simples e específico para uma tabela, enquanto a sequence oferece mais flexibilidade, permitindo controle sobre o valor inicial, incremento e até mesmo o uso em várias tabelas.

## **Qual a importância das chaves estrangeiras para a consistência do banco?**

As chaves estrangeiras são essenciais para garantir a consistência do banco de dados, pois elas asseguram que os dados em uma tabela estejam relacionados corretamente com os dados em outra tabela. Elas evitam que registros sejam inseridos sem um vínculo válido, impedindo, por exemplo, que uma venda seja registrada sem um cliente associado ou que um pedido tenha um produto inexistente. Além disso, as chaves estrangeiras ajudam a manter a integridade referencial, garantindo que, se um registro de uma tabela for alterado ou excluído, as tabelas relacionadas sejam atualizadas de forma consistente. Isso evita dados órfãos e incoerentes no banco de dados.

## **Quais operadores do SQL pertencem à álgebra relacional e quais são definidos no cálculo relacional?**

No SQL, os operadores da álgebra relacional incluem select, join, union, intersect, except e cross join, que manipulam diretamente as tabelas. Já os operadores do cálculo relacional incluem exists, not exists, in, any e all, que são usados para expressar condições lógicas mais complexas e determinar resultados com base em predicados.

## **Como é feito o agrupamento em consultas, e qual requisito é obrigatório?**

O agrupamento em consultas no SQL é feito utilizando a cláusula group by, que organiza os resultados em grupos com base em uma ou mais colunas. O requisito obrigatório ao usar group by é que todas as colunas na lista de seleção (exceto as funções agregadas como count, sum, avg, etc.) devem estar incluídas no group by, garantindo que os dados sejam agrupados corretamente. Isso assegura que os resultados sejam apresentados de forma coerente para cada grupo.