



Campus: POLO SAGUAÇU - JOINVILLE - SC

Curso: DESENVOLVIMENTO FULL STACK

Disciplina: Vamos Manter as Informações?

Turma: 9001

Semestre: 1º Semestre (2024)

Aluno: Jederson Borges de Oliveira

Link: <https://github.com/JedersonBorges/missao-pratica-nivel-2-mundo-3.git>

## Missão Prática | Nível 2 | Mundo 3

### Objetivos da prática

1. Identificar os requisitos de um sistema e transformá-los no modelo adequado.
2. Utilizar ferramentas de modelagem para bases de dados relacionais.
3. Explorar a sintaxe SQL na criação das estruturas do banco (DDL).
4. Explorar a sintaxe SQL na consulta e manipulação de dados (DML).
5. 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

Inserindo dados na tabela **Usuarios** (Peço desculpas se entendi errado, mas não foi solicitado colocar login e senha quando criava diagrama no DBDesigner, por esse motivo também não inseri os campos aqui)

The screenshot shows a SQL query window in SQL Server Enterprise Manager. The query is as follows:

```
INSERT INTO Usuarios (idUsuario, nome, email, nome_2)
VALUES
(1, 'op1', 'op1', ''),
(2, 'op2', 'op2', '');

SELECT * FROM Usuarios;
```

Below the query, the results are displayed in a table with 7 columns: idUsuario, pessoaJuridica\_cnpj, pessoaFisica\_cpf, nome, email, and nome\_2. The results show two rows of data.

|   | idUsuario | pessoaJuridica_cnpj | pessoaFisica_cpf | nome | email | nome_2 |
|---|-----------|---------------------|------------------|------|-------|--------|
| 1 | 1         | NULL                | NULL             | op1  | op1   |        |
| 2 | 2         | NULL                | NULL             | op2  | op2   |        |

### Inserindo Produtos na tabela:

SQLQueryProdutos.sql...16.Loja (loja (61))

```
INSERT INTO Produto (idProduto, nome, quantidade, precoVenda)
VALUES
(1, 'Banana', 100, 5.00),
(3, 'Laranja', 500, 2.00),
(4, 'Manga', 800, 4.00);

SELECT * FROM Produto;
```

91 %

Resultados Mensagens

|   | idProduto | nome    | quantidade | precoVenda |
|---|-----------|---------|------------|------------|
| 1 | 1         | Banana  | 100        | 5          |
| 2 | 3         | Laranja | 500        | 2          |
| 3 | 4         | Manga   | 800        | 4          |

### Inserindo Pessoas Físicas e Jurídicas:

SQLQueryPessoas.sql...16.Loja (loja (61))

```
-- Inserir pessoas físicas
INSERT INTO Pessoas (idPessoa, nome, logradouro, cidade, estado, telefone, email)
VALUES
(7, 'Joao', 'Rua 12, casa 3, Quitanda', 'Riacho do Sul', 'PA', '1111-1111', 'joao@riacho.com');

INSERT INTO PessoaFisica (cpf, idPessoa)
VALUES
(11111111111, 7);

-- Inserir pessoas jurídicas
INSERT INTO Pessoas (idPessoa, nome, logradouro, cidade, estado, telefone, email)
VALUES
(15, 'JJC', 'Rua 11, Centro', 'Riacho do Norte', 'PA', '1212-1212', 'jcc@riacho.com');

INSERT INTO PessoaJuridica (cnpj, idPessoa)
VALUES
(2222222222222, 15);

SELECT * FROM Pessoas;
SELECT * FROM pessoaJuridica;
SELECT * FROM pessoaFisica;
```

91 %

Resultados Mensagens

|   |    | Joao | Rua 12, casa 3, Quitanda | Riacho do Sul   | PA | 1111-1111 | joao@riacho.com |
|---|----|------|--------------------------|-----------------|----|-----------|-----------------|
| 2 | 15 | JJC  | Rua 11, Centro           | Riacho do Norte | PA | 1212-1212 | jcc@riacho.com  |

|   | cnpj          | idPessoa |
|---|---------------|----------|
| 1 | 2222222222222 | 15       |

|   | cpf         | idPessoa |
|---|-------------|----------|
| 1 | 11111111111 | 7        |

## Inserindo Movimentos

SQLQueryMovimento...6.Loja (loja (60))

```
INSERT INTO Movimento (idMovimento, Usuarios_idUsuario, idPessoa, idProduto, quantidade, tipo, valorUnitario)
VALUES
(1, 1, 7, 1, 20, 'S', 4.00),
(4, 1, 7, 3, 15, 'S', 2.00),
(5, 2, 7, 3, 10, 'S', 3.00),
(7, 1, 15, 3, 15, 'E', 5.00),
(8, 1, 15, 4, 20, 'E', 4.00);

SELECT * FROM Movimento;
```

91 %

Resultados Mensagens

|   | idMovimento | Usuarios_idUsuario | idPessoa | idProduto | quantidade | tipo | valorUnitario |
|---|-------------|--------------------|----------|-----------|------------|------|---------------|
| 1 | 1           | 1                  | 7        | 1         | 20         | S    | 4             |
| 2 | 4           | 1                  | 7        | 3         | 15         | S    | 2             |
| 3 | 5           | 2                  | 7        | 3         | 10         | S    | 3             |
| 4 | 7           | 1                  | 15       | 3         | 15         | E    | 5             |
| 5 | 8           | 1                  | 15       | 4         | 20         | E    | 4             |

## Consultas solicitadas:

### a) Dados completos pessoas físicas:

SQLQuery24.sql - JE...N16.Loja (loja (73))\*

```
SELECT p.idPessoa, p.nome, p.logradouro, p.cidade, p.estado, p.telefone, p.email, pf.cpf
FROM Pessoas p
JOIN PessoaFisica pf ON p.idPessoa = pf.idPessoa;
```

91 %

Resultados Mensagens

|   | idPessoa | nome | logradouro               | cidade        | estado | telefone  | email           | cpf         |
|---|----------|------|--------------------------|---------------|--------|-----------|-----------------|-------------|
| 1 | 7        | Joao | Rua 12, casa 3, Quitanda | Riacho do Sul | PA     | 1111-1111 | joao@riacho.com | 11111111111 |

### b) Dados completos pessoas jurídicas:

SQLQueryDadosCompl...6.Loja (loja (60))

```
SELECT p.idPessoa, p.nome, p.logradouro, p.cidade, p.estado, p.telefone, p.email, pj.cnpj
FROM Pessoas p
JOIN PessoaJuridica pj ON p.idPessoa = pj.idPessoa;
```

91 %

Resultados Mensagens

|   | idPessoa | nome | logradouro     | cidade          | estado | telefone  | email          | cnpj           |
|---|----------|------|----------------|-----------------|--------|-----------|----------------|----------------|
| 1 | 15       | JJC  | Rua 11, Centro | Riacho do Norte | PA     | 1212-1212 | jcc@riacho.com | 22222222222222 |

### c) Movimentações de entrada, com produto, fornecedor, quantidade, preço unitário e valor total:

SQLQueryDadosCompl...6.Loja (loja (75))

```

SELECT m.idMovimento, p.idPessoa AS fornecedor, prod.nome AS produto, m.quantidade, m.valorUnitario, (m.quantidade * m.valorUnitario) AS valorTotal
FROM Movimento m
JOIN Pessoas p ON m.idPessoa = p.idPessoa
JOIN Produto prod ON m.idProduto = prod.idProduto
WHERE m.tipo = 'E';

```

91 %

|   | idMovimento | fornecedor | produto | quantidade | valorUnitario | valorTotal |
|---|-------------|------------|---------|------------|---------------|------------|
| 1 | 7           | 15         | Laranja | 15         | 5             | 75         |
| 2 | 8           | 15         | Manga   | 20         | 4             | 80         |

**d) Movimentações de saída, com produto, comprador, quantidade, preço unitário e valor total:**

SQLQueryDadosCompl...6.Loja (loja (60))

```

SELECT m.idMovimento, p.idPessoa AS comprador, prod.nome AS produto, m.quantidade, m.valorUnitario, (m.quantidade * m.valorUnitario) AS valorTotal
FROM Movimento m
JOIN Pessoas p ON m.idPessoa = p.idPessoa
JOIN Produto prod ON m.idProduto = prod.idProduto
WHERE m.tipo = 'S';

```

91 %

|   | idMovimento | comprador | produto | quantidade | valorUnitario | valorTotal |
|---|-------------|-----------|---------|------------|---------------|------------|
| 1 | 1           | 7         | Banana  | 20         | 4             | 80         |
| 2 | 4           | 7         | Laranja | 15         | 2             | 30         |
| 3 | 5           | 7         | Laranja | 10         | 3             | 30         |

**e) Valor total das entradas agrupadas por produto:**

SQLQueryTotalEntrad...16.Loja (loja (69))

```

SELECT prod.nome AS produto, SUM(m.quantidade * m.valorUnitario) AS valorTotalEntrada
FROM Movimento m
JOIN Produto prod ON m.idProduto = prod.idProduto
WHERE m.tipo = 'E'
GROUP BY prod.nome;

```

91 %

|   | produto | valorTotalEntrada |
|---|---------|-------------------|
| 1 | Laranja | 75                |
| 2 | Manga   | 80                |

**f) Valor total das saídas agrupadas por produto:**

SQLQueryTotalSaidas...16.Loja (loja (75))

```

SELECT prod.nome AS produto, SUM(m.quantidade * m.valorUnitario) AS valorTotalSaida
FROM Movimento m
JOIN Produto prod ON m.idProduto = prod.idProduto
WHERE m.tipo = 'S'
GROUP BY prod.nome;

```

91 %

|   | produto | valorTotalSaida |
|---|---------|-----------------|
| 1 | Banana  | 80              |
| 2 | Laranja | 60              |

**g) Operadores que não efetuaram movimentações de entrada (compra):**

SQLQueryOperadores...16.Loja (loja (69))

```

SELECT u.idUsuario, u.nome
FROM Usuarios u
LEFT JOIN Movimento m ON u.idUsuario = m.Usuarios_idUsuario AND m.tipo = 'E'
WHERE m.idMovimento IS NULL;

```

91 %

Resultados Mensagens

|   | idUsuario | nome |
|---|-----------|------|
| 1 | 2         | op2  |

h) Valor total de entrada, agrupado por operador:

SQLQueryValorTotalE...16.Loja (loja (75))

```

SELECT u.idUsuario, u.nome, SUM(m.quantidade * m.valorUnitario) AS valorTotalEntrada
FROM Usuarios u
JOIN Movimento m ON u.idUsuario = m.Usuarios_idUsuario
WHERE m.tipo = 'E'
GROUP BY u.idUsuario, u.nome;

```

91 %

Resultados Mensagens

|   | idUsuario | nome | valorTotalEntrada |
|---|-----------|------|-------------------|
| 1 | 1         | op1  | 155               |

i) Valor total de saída, agrupado por operador:

SQLQueryValorTotalS...16.Loja (loja (75))

```

SELECT u.idUsuario, u.nome, SUM(m.quantidade * m.valorUnitario) AS valorTotalSaida
FROM Usuarios u
JOIN Movimento m ON u.idUsuario = m.Usuarios_idUsuario
WHERE m.tipo = 'S'
GROUP BY u.idUsuario, u.nome;

```

91 %

Resultados Mensagens

|   | idUsuario | nome | valorTotalSaida |
|---|-----------|------|-----------------|
| 1 | 1         | op1  | 110             |
| 2 | 2         | op2  | 30              |

j) valor médio de venda por produto, utilizando média ponderada:

SQLQueryValorMedio...16.Loja (loja (75))

```
SELECT prod.nome AS produto,
      SUM(m.quantidade * m.valorUnitario) / SUM(m.quantidade) AS mediaPonderada
FROM Movimento m
JOIN Produto prod ON m.idProduto = prod.idProduto
WHERE m.tipo = 'S'
GROUP BY prod.nome;
```

91 %

Resultados

Mensagens

|   | produto | mediaPonderada |
|---|---------|----------------|
| 1 | Banana  | 4.000000       |
| 2 | Laranja | 2.400000       |

## Análise e Conclusão

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

Identity é utilizada para maior simplicidade em uma única tabela, enquanto Sequence é usada quando precisamos de mais controle e quando precisamos compartilhar valores entre várias tabelas

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

As chaves estrangeiras são cruciais para a integridade referencial do banco de dados. Elas garantem que os relacionamentos entre tabelas sejam mantidos corretamente.

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

Os operadores do SQL que pertencem à álgebra relacional incluem seleção, projeção, união, diferença, produto cartesiano, junção e renomeação, enquanto o cálculo relacional usa expressões de predicado para definir consultas.

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

O agrupamento em consultas SQL é feito usando GROUP BY. Isso agrupa linhas com valores comuns em colunas específicas. Depois, você pode usar funções como COUNT, SUM, AVG, etc., para calcular resultados para cada grupo.

O requisito obrigatório é que todas as colunas na cláusula SELECT, que não estão em uma função de agregação, precisam estar na cláusula GROUP BY.

