



Campus: Madureira

Curso: Desenvolvimento full stack

Disciplina: Vamos manter as informações?

Turma: 9001

Semestre: 3º

Nome: Lincoln Miguel da Cruz Antonio

Missão Prática - Nível 2

Todos os códigos usados:

```
SELECT *  
FROM PessoaFisica
```

```
SELECT *  
FROM PessoaJuridica
```

```
SELECT  
    m.idMovimento AS MovimentacaoID,  
    p.nome AS Produto,  
    pj.razaoSocial AS Fornecedor,  
    m.quantidadeProduto AS Quantidade,  
    m.valorUnitario AS Preço_Unitário,  
    (m.quantidadeProduto * m.valorUnitario) AS Valor_Total
```

```
FROM Movimento m
JOIN Produto p ON m.idMovimento =
p.Movimento_idMovimento
JOIN PessoaJuridica pj ON
m.PessoaJuridica_idPessoaJuridica = pj.idPessoaJuridica
```

```
SELECT
    m.idMovimento AS MovimentacaoID,
    p.nome AS Produto,
    pf.nome AS Comprador,
    m.quantidadeProduto AS Quantidade,
    m.valorUnitario AS Preço_Unitario,
    (m.quantidadeProduto * m.valorUnitario) AS Valor_Total
FROM Movimento m
JOIN Produto p ON m.idMovimento =
p.Movimento_idMovimento
JOIN PessoaFisica pf ON m.PessoaFisica_idPessoaFisica =
pf.idPessoaFisica
```

```
SELECT
    p.nome AS Produto,
    SUM(m.quantidadeProduto * m.valorUnitario) AS
Valor_Total
FROM Movimento m
JOIN Produto p ON m.idMovimento =
p.Movimento_idMovimento
GROUP BY p.nome;
```

```
SELECT
    p.nome AS Produto,
    SUM(m.quantidadeProduto * m.valorUnitario) AS
Valor_Total
FROM Movimento m
```

```
JOIN Produto p ON m.idMovimento =  
p.Movimento_idMovimento  
GROUP BY p.nome;
```

```
SELECT u.*  
FROM Usuario u  
WHERE NOT EXISTS (  
    SELECT 1  
    FROM Movimento m  
        WHERE m.Usuario_idUsuario = u.idUsuario AND m.tipo  
= 'compra'  
);
```

```
SELECT  
    u.login AS Operador,  
    SUM(m.quantidadeProduto * m.valorUnitario) AS  
Valor_Total  
FROM Movimento m  
JOIN Usuario u ON m.Usuario_idUsuario = u.idUsuario  
GROUP BY u.login;
```

```
SELECT  
    u.login AS Operador,  
    SUM(m.quantidadeProduto * m.valorUnitario) AS  
Valor_Total  
FROM Movimento m  
JOIN Usuario u ON m.Usuario_idUsuario = u.idUsuario  
GROUP BY u.login;
```

```
SELECT  
    p.nome AS Produto,
```

SUM(m.quantidadeProduto * m.valorUnitario) /
 NULLIF(SUM(m.quantidadeProduto), 0) AS
 Media_Ponderada
 FROM Movimento m
 JOIN Produto p ON m.idMovimento =
 p.Movimento_idMovimento
 GROUP BY p.nome;

Resultados dos códigos:

The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The left pane displays the 'Loja' database structure. The right pane shows a SQL query and its results.

Query:

```

SELECT
FROM PessoaFisica
FROM PessoaJuridica
SELECT
m.idMovimento AS MovimentoID,
p.nome AS Produto,
pj.razaSocial AS Fornecedor,
m.quantidadeProduto AS Quantidade,
m.valorUnitario AS Preço Unitário,
(m.quantidadeProduto * m.valorUnitario) AS Valor Total
FROM Movimento m
JOIN Produto p ON m.idMovimento = p.Movimento_idMovimento
JOIN PessoaJuridica pj ON m.PessoaJuridica_idPessoaJuridica = pj.idPessoaJuridica
SELECT
m.idMovimento AS MovimentoID,
p.nome AS Produto
  
```

Results:

| idPessoaFisica | nome | cpf | logradouro | cidade | telefone | email |
|----------------|------|--------------|---------------|----------------|------------|--------------|
| 1 | Leo | 111111111111 | Rua 5, centro | Rio de Janeiro | 2222222222 | leo@dmas.com |

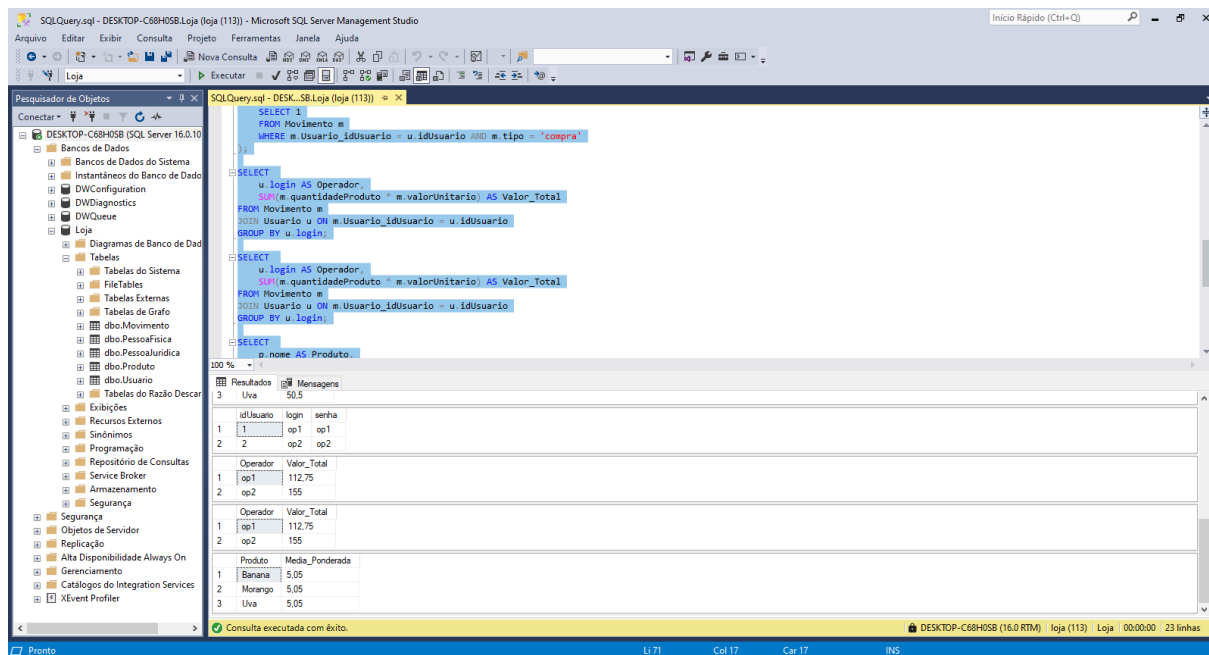
| idPessoaJuridica | razaSocial | cnpj | logradouro | cidade | telefone | email |
|------------------|------------|----------------|---------------|----------------|------------|------------------|
| 2 | Empresa x | 33333333333333 | Rua 10, praça | Rio de Janeiro | 5555555555 | empresa@foco.com |

| MovimentoID | Produto | Fornecedor | Quantidade | Preço Unitário | Valor Total |
|-------------|---------|------------|------------|----------------|-------------|
| 3 | Uva | Empresa x | 10 | 5,05 | 50,5 |
| 3 | Banana | Empresa x | 10 | 5,05 | 50,5 |
| 3 | Morango | Empresa x | 10 | 5,05 | 50,5 |

| MovimentoID | Produto | Comprador | Quantidade | Preço Unitário | Valor Total |
|-------------|---------|-----------|------------|----------------|-------------|
| 3 | Uva | Leo | 10 | 5,05 | 50,5 |
| 3 | Banana | Leo | 10 | 5,05 | 50,5 |
| 3 | Morango | Leo | 10 | 5,05 | 50,5 |

| Produto | Valor Total |
|---------|-------------|
| Banana | 50,5 |

Consulta executada com êxito.



Análise e Conclusão:

a. Quais as diferenças no uso de sequence e identity?

Sequence não está preso a nenhuma tabela, já identity faz a geração de dados para uma tabela.

b. Qual a importância das chaves estrangeiras para a consistência do banco?

Elas ajudam a proteger os dados, de forma que você não pode modificá-los, excluir, atualizar e etc...

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

União - Faz união entre resultados de tabelas.

Projeção - Faz um filtro entre colunas de tabelas.

Diferença - Retorna linhas diferentes entre tabelas.

Produto cartesiano - Retorna um terceiro resultado combinado do primeiro e segundo.

Junção - Faz combinação entre linhas de tabelas.

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

Através do group by e é usado para juntar resultados de colunas e aplicar funções.