



Estácio

Faculdade Estácio

Campus Belford Roxo - RJ

Desenvolvimento Full Stack

Disciplina - Vamos manter as informações

Turma 2023.2

Semestre - 3

Cristian da Silva de Macena

Código

--Criação da tabela usuario

```
CREATE TABLE usuario(  
    idUsuario INT PRIMARY KEY IDENTITY(1,1) NOT NULL,  
    login VARCHAR(255) NOT NULL,  
    senha VARCHAR(255) NOT NULL  
)
```

--Criação da tabela produtos

```
CREATE TABLE produtos(  
    idProduto INT PRIMARY KEY IDENTITY(1,1) NOT NULL,  
    nome VARCHAR(255) NOT NULL,  
    quantidade INT NOT NULL,  
    precoVenda NUMERIC NOT NULL,  
)
```

--Criação da tabela movimento

```
CREATE TABLE movimento(  
    idMovimento INT PRIMARY KEY IDENTITY(1,1) NOT NULL,  
    idUsuario INT NULL,  
    idPessoa INT NULL,  
    idProduto INT NULL,  
    quantidade INT NULL,  
    tipo CHAR(1) NULL,  
    valorUnitario NUMERIC NULL,  
    FOREIGN KEY (idUsuario) REFERENCES usuario(idUsuario),  
    FOREIGN KEY (idPessoa) REFERENCES pessoa(idPessoa),  
    FOREIGN KEY (idProduto) REFERENCES produtos(idProduto)  
)
```

--Inserindo um registro na tabela pessoa

```
INSERT INTO pessoa(nome,logradouro,cidade,estado,telefone,email) VALUES  
('Lucas','Rua Maria','Nova iguaçu','RJ','1111-1111','email@gmail.com')
```

--Inserindo o id número 1 da tabela pessoa como pessoa física

```
INSERT INTO pessoaFisica(idPessoa_Fk,cpf) VALUES (1,'11111111111')
```

--Inserindo um registro na tabela pessoa

```
INSERT INTO pessoa(nome,logradouro,cidade,estado,telefone,email) VALUES ('Ana','Rua  
Santos','Caxias','RJ','2222-2222','email@hotmail.com')
```

--Inserindo o id número 2 da tabela pessoa como pessoa jurídica

```
INSERT INTO pessoaJuridica(idPessoa_Fk,cnpj) VALUES (2,'2222222222222')
```

--Dados completos de pessoa jurídica

```
SELECT * FROM pessoa p INNER JOIN pessoaJuridica pj ON p.idPessoa = pj.idPessoa_Fk
```

--Dados completos de pessoa fisica

```
SELECT * FROM pessoa p INNER JOIN pessoaFisica pf ON p.idPessoa = pf.idPessoa_Fk
```

--Inserindo dados na tabela movimento

```
INSERT INTO movimento(idPessoa,idProduto,idUsuario,quantidade,tipo,valorUnitario)
VALUES (1,1,1,20,'E',20.10)
```

--Dados de entrada

```
SELECT * FROM movimento m WHERE m.tipo = 'E'
```

--Dados de saída

```
SELECT * FROM movimento m WHERE m.tipo = 'S'
```

--Valor das entradas

```
SELECT * FROM movimento m WHERE m.tipo = 'E' ORDER BY m.idProduto ASC
```

--Valor das saídas

```
SELECT * FROM movimento m WHERE m.tipo = 'S' ORDER BY m.idProduto ASC
```

--Operadores que não efetuaram movimentações de entrada

```
SELECT * FROM movimento m WHERE m.tipo != 'E'
```

--Valor total de entrada

```
SELECT SUM(m.valorUnitario) valorTotal FROM movimento m WHERE m.tipo = 'E'
```

--Valor total de saída

```
SELECT SUM(m.valorUnitario) valorTotal FROM movimento m WHERE m.tipo = 'S'
```

Execução

150 %

Resultados Mensagens

	idPessoa	nome	logradouro	cidade	estado	telefone	email	idPessoa_Juridica	idPessoa_Fk	cnpj	
1	2	Ana	Rua Santos	Caxias	RJ	2222-2222	email@hotmail.com	1	2	22222222222222	

	idPessoa	nome	logradouro	cidade	estado	telefone	email	idPessoa_Fisica	idPessoa_Fk	cpf	
1	1	Lucas	Rua Maria	Nova iguaçu	RJ	1111-1111	email@gmail.com	1	1	11111111111	

150 %

Resultados		Mensagens					
	idMovimento	idUsuario	idPessoa	idProduto	quantidade	tipo	valorUnitario
1	2	1	1	1	10	S	21
2	3	1	1	1	20	E	20
3	7	1	1	2	20	E	20

Quais as diferenças no uso de sequence e identity?

SEQUENCE é um objeto independente que gera valores sequenciais e pode ser chamado diretamente por aplicativos, enquanto IDENTITY está vinculado a uma tabela específica e não altera registros anteriores, sendo útil para transações de INSERT.

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

A chave estrangeira é importante para garantir a integridade dos dados e fazer o relacionamento entre tabelas.

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

Na álgebra relacional, temos operadores como seleção, projeção, produto cartesiano, união e diferença entre conjuntos. Já o cálculo relacional é declarativo e não possui operadores específicos, sendo usado para expressar consultas de maneira intuitiva.

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

O agrupamento em consultas SQL é realizado usando a cláusula GROUP BY. Essa cláusula nos permite agrupar registros com base em dados de uma coluna específica (ou colunas).