

# Missao Pratica | Nivel 2 | Mundo 3



**Universidade: Estácio**

**Campus: 125 POLO R9 - TAQUARA - RJ**

**Curso: Desenvolvimento Full Stack**

**Disciplina: Nível 2: Vamos Manter as Informações?**

**Turma: 9001**

**Semestre Letivo: Terceiro**

**Nome dos integrantes: Marlon Santos Corrêa**

**<https://github.com/marlonsc15/Missao-Pratica-Nivel-2-Mundo-3.git>**

## 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.**

# Resultados da Execucao

## Dados completos de pessoas físicas

<pre>SELECT * FROM pessoa p INNER JOIN pessoaFisica pf ON p.id_pessoa = pf.id_pessoa;</pre>									
100 %									
Resultados Mensagens									
	id_pessoa	nome	email	telefone	estado	logradouro	cidade	id_pessoa	cpf
1	2	Joao	joao@riacho.com	1111-1111	Rua 12, casa 3, Quitanda	NULL	NULL	2	11111111111
2	6	Maria Silva	maria@email.com	(19)99999-0000	SP	Rua das Rosas	Campinas	6	12345678901

## Dados completos de pessoas jurídicas

<pre>SELECT * FROM pessoa p INNER JOIN PessoaJuridica pj ON p.id_pessoa = pj.id_pessoa;</pre>									
100 %									
Resultados Mensagens									
	id_pessoa	nome	email	telefone	estado	logradouro	cidade	id_pessoa	cnpj
1	7	Loja ABC Ltda	contato@lojaabc.com	(11)98888-1234	SP	Av. Central	São Paulo	7	12345678000199
2	10	Loja ABC Ltda	contato@lojaabc.com	(11)98888-1234	SP	Av. Central	São Paulo	10	12345678000599
3	3	JJC	jjc@riacho.com	1212-1212	Rua 11, Centro	NULL	NULL	3	22222222222222

Movimentações de entrada, com produto, fornecedor,  
quantidade, preço unitário e valor total

```
SELECT
  c.id_compra,
  pr.nome AS produto,
  pj.cnpj AS fornecedor_cnpj,
  p.nome AS fornecedor_nome,
  c.quantidade,
  c.preco_unitario,
  (c.quantidade * c.preco_unitario) AS valor_total,
  c.data_compra
FROM
  compra c
INNER JOIN
  produto pr ON c.id_produto = pr.id_produto
INNER JOIN
  pessoaJuridica pj ON c.id_pessoa_jur = pj.id_pessoa
INNER JOIN
  pessoa p ON pj.id_pessoa = p.id_pessoa;
```

100 %

Resultados Mensagens

	id_compra	produto	fomecedor_cnpj	fomecedor_nome	quantidade	preco_unitario	valor_total	data_compra
1	4	Banana	22222222222222	JJC	50	2.50	125.00	2025-04-22

Movimentações de saída, com produto, comprador,  
quantidade, preço unitário e valor total

```
SELECT
  v.id_venda,
  pr.nome AS produto,
  pf.cpf AS comprador_cpf,
  p.nome AS comprador_nome,
  v.quantidade,
  v.preco_unitario,
  (v.quantidade * v.preco_unitario) AS valor_total,
  v.data_venda
FROM
  venda v
INNER JOIN
  produto pr ON v.id_produto = pr.id_produto
INNER JOIN
  pessoaFisica pf ON v.id_pessoa_fis = pf.id_pessoa
INNER JOIN
  pessoa p ON pf.id_pessoa = p.id_pessoa;
```

100 %

Resultados Mensagens

	id_venda	produto	comprador_cpf	comprador_nome	quantidade	preco_unitario	valor_total	data_venda
1	1	Banana	111111111111	Joao	10	4.50	45.00	2025-04-22 16:07:36.073

### Valor total das entradas agrupadas por produto

```
SELECT
    pr.nome AS produto,
    SUM(c.quantidade * c.preco_unitario) AS valor_total_entrada
FROM
    compra c
INNER JOIN
    produto pr ON c.id_produto = pr.id_produto
GROUP BY
    pr.nome;
```

100 %

Resultados Mensagens

	produto	valor_total_entrada
1	Banana	125.00

### Valor total das saídas agrupadas por produto

```
SELECT
    pr.nome AS produto,
    SUM(v.quantidade * v.preco_unitario) AS valor_total_saida
FROM
    venda v
INNER JOIN
    produto pr ON v.id_produto = pr.id_produto
GROUP BY
    pr.nome;
```

100 %

Resultados Mensagens

	produto	valor_total_saida
1	Banana	45.00

Operadores que não efetuaram movimentações de entrada (compra)

```
SELECT
    u.id_usuario,
    u.login,
    u.senha
FROM
    usuario u
WHERE
    u.id_usuario NOT IN (
        SELECT DISTINCT id_usuario FROM compra
    );
```

100 %

Resultados Mensagens

	id_usuario	login	senha
1	3	op2	op2

Valor total de entrada, agrupado por operador

```
SELECT
    u.id_usuario AS operador,
    SUM(c.quantidade * c.preco_unitario) AS valor_total_entrada
FROM
    compra c
INNER JOIN
    usuario u ON c.id_usuario = u.id_usuario
GROUP BY
    u.id_usuario;
```

100 %

Resultados Mensagens

	operador	valor_total_entrada
1	2	125.00

### Valor total de saída, agrupado por operador

```
SELECT
    u.id_usuario AS operador,
    SUM(v.quantidade * v.preco_unitario) AS valor_total_saida
FROM
    venda v
INNER JOIN
    usuario u ON v.id_usuario = u.id_usuario
GROUP BY
    u.id_usuario;
```

100 %

Resultados Mensagens

	operador	valor_total_saida
1	2	45.00

### Valor médio de venda por produto, utilizando média ponderada

```
SELECT
    p.nome AS produto,
    CAST(SUM(v.quantidade * v.preco_unitario) AS FLOAT) / NULLIF(SUM(v.quantidade), 0) AS media_ponderada
FROM
    venda v
INNER JOIN
    produto p ON v.id_produto = p.id_produto
GROUP BY
    p.nome;
```

100 %

Resultados Mensagens

	produto	media_ponderada
1	Banana	4.5

## **Analise e Conclusao**

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

**R:** IDENTITY é usado diretamente em uma coluna de tabela e SEQUENCE é um objeto separado do banco, que pode ser usado em várias tabelas.

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

**R:** São essenciais para manter os relacionamentos corretos entre tabelas e evitar erros lógicos nos dados.

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

**R:** SELECT, WHERE, GROUP BY, ORDER BY, UNION, INTERSECT, EXCEPT, JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN e CROSS JOIN são Álgebra Relacional.

Ja =, >, <, !=, AND, OR, NOT são Calculo Relacional.

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

**R:** O agrupamento em consultas é feito com a cláusula GROUP BY no SQL, e o requisito obrigatório é que se houver funções de agregação (SUM, AVG, COUNT, MAX, MIN) na consulta, todas as colunas não agregadas que estão sendo exibidas na seleção ( SELECT ) também precisam estar na cláusula GROUP BY.