

Trabalho de Banco de Dados - Projeto – Parte II (Continuação parte I)

Melhor visualização acesse: https://github.com/r-milioli/avaliacao-banco-de-dados/blob/main/FRISTORM_INFORMATICA/BANCO_DADOS_FOLDER/parte2_consultas_comp letas.sql

Tabela de Avaliação por Critério

```
-- =====
-- PARTE 2 - CONSULTAS E OPERAÇÕES AVANÇADAS EM SQL
-- FRISTORM INFORMÁTICA - Sistema de Gestão de Estoque e Vendas
-- =====

-- Certifique-se de estar usando o banco correto
USE fristorm_informatica;

-- =====
-- CRITÉRIO 1: 2 consultas com SELECT e WHERE (2,0 pts)
-- =====

-- Consulta 1.1 - Produtos com quantidade em estoque maior que 100
SELECT codProduto, nomeProduto
FROM Produto
WHERE codProduto IN (
    SELECT DISTINCT codProduto
    FROM estoque
    WHERE codEstoque IN (
        SELECT codEstoque
        FROM Estoque
        WHERE qdtProduto > 100
    )
);
```

-- Consulta 1.2 - Fornecedores que fornecem produtos específicos

```
SELECT codFornecedor, nomeFornecedor
FROM Fornecedor
WHERE codFornecedor IN (
    SELECT codFornecedor
    FROM fornece
    WHERE codProduto IN (1, 2, 3)
);
```

-- =====

-- CRITÉRIO 2: 2 consultas com GROUP BY e ORDER BY com funções de agregação (2,0 pts)

-- =====

-- Consulta 2.1 - Total de produtos por fornecedor

```
SELECT f.nomeFornecedor, COUNT(fo.codProduto) as Total_Produtos
FROM Fornecedor f
JOIN fornece fo ON f.codFornecedor = fo.codFornecedor
GROUP BY f.codFornecedor, f.nomeFornecedor
ORDER BY Total_Produtos DESC;
```

-- Consulta 2.2 - Média de quantidade de compra por cliente

```
SELECT c.codCliente, AVG(c.qdtCompra) as Media_Compra
FROM Cliente c
GROUP BY c.codCliente
ORDER BY Media_Compra DESC;
```

-- =====

-- CRITÉRIO 3: 2 consultas com operadores aritméticos (+, -, *, /) (3,0 pts)

-- =====

-- Consulta 3.1 - Cálculo de estoque disponível por produto

```
SELECT p.codProduto, p.nomeProduto,  
       (e.qdtProduto * 0.8) as Estoque_Disponivel,  
       (e.qdtProduto * 0.2) as Estoque_Reserva  
FROM Produto p  
JOIN estoque es ON p.codProduto = es.codProduto  
JOIN Estoque e ON es.codEstoque = e.codEstoque  
WHERE e.qdtProduto > 50;
```

-- Consulta 3.2 - Análise de compras com desconto

```
SELECT c.codCliente, c.qdtCompra,  
       (c.qdtCompra * 10) as Valor_Original,  
       (c.qdtCompra * 10 * 0.9) as Valor_Com_Desconto,  
       (c.qdtCompra * 10 * 0.1) as Desconto_Aplicado  
FROM Cliente c  
WHERE c.qdtCompra > 5;
```

-- =====

-- CRITÉRIO 4: 3 consultas com operadores de comparação (=, !=, <, >, etc.) (2,0 pts)

-- =====

-- Consulta 4.1 - Produtos com estoque específico

```
SELECT p.nomeProduto, e.qdtProduto  
FROM Produto p  
JOIN estoque es ON p.codProduto = es.codProduto  
JOIN Estoque e ON es.codEstoque = e.codEstoque  
WHERE e.qdtProduto = 150;
```

-- Consulta 4.2 - Clientes com quantidade de compra diferente de 5

```
SELECT codCliente, qdtCompra  
FROM Cliente
```

WHERE qdtCompra != 5;

-- Consulta 4.3 - Fornecedores com mais de 2 produtos

```
SELECT f.nomeFornecedor, COUNT(fo.codProduto) as Total_Produtos
FROM Fornecedor f
JOIN fornece fo ON f.codFornecedor = fo.codFornecedor
GROUP BY f.codFornecedor
HAVING COUNT(fo.codProduto) > 2;
```

-- =====

-- CRITÉRIO 5: 3 consultas com operadores lógicos (AND, OR) (3,0 pts)

-- =====

-- Consulta 5.1 - Produtos em estoques específicos

```
SELECT p.nomeProduto, e.codEstoque, e.qdtProduto
FROM Produto p
JOIN estoque es ON p.codProduto = es.codProduto
JOIN Estoque e ON es.codEstoque = e.codEstoque
WHERE e.codEstoque = 1 AND e.qdtProduto > 100;
```

-- Consulta 5.2 - Clientes com compras específicas

```
SELECT codCliente, qdtCompra
FROM Cliente
WHERE qdtCompra = 3 OR qdtCompra = 8;
```

-- Consulta 5.3 - Fornecedores e produtos específicos

```
SELECT f.nomeFornecedor, p.nomeProduto
FROM Fornecedor f
JOIN fornece fo ON f.codFornecedor = fo.codFornecedor
JOIN Produto p ON fo.codProduto = p.codProduto
WHERE f.codFornecedor = 1 AND p.codProduto IN (1, 2, 3);
```

-- =====

-- *CRITÉRIO 6: 2 consultas com operadores lógicos e negação (NOT) (3,0 pts)*

-- =====

-- *Consulta 6.1 - Produtos não fornecidos por fornecedor específico*

```
SELECT p.nomeProduto
FROM Produto p
WHERE p.codProduto NOT IN (
    SELECT codProduto
    FROM fornece
    WHERE codFornecedor = 1
);
```

-- *Consulta 6.2 - Clientes que não compraram produtos específicos*

```
SELECT c.codCliente, c.qdtCompra
FROM Cliente c
WHERE c.codCliente NOT IN (
    SELECT codCliente
    FROM adquiere
    WHERE codProduto IN (1, 2)
);
```

-- =====

-- *CRITÉRIO 7: 3 consultas com operadores auxiliares (IS NULL, BETWEEN, LIKE, IN) (3,0 pts)*

-- =====

-- *Consulta 7.1 - Produtos com nomes que contêm 'Dell'*

```
SELECT codProduto, nomeProduto
FROM Produto
WHERE nomeProduto LIKE '%Dell%';
```

-- Consulta 7.2 - Estoques com quantidade entre 100 e 200

```
SELECT codEstoque, qdtProduto
FROM Estoque
WHERE qdtProduto BETWEEN 100 AND 200;
```

-- Consulta 7.3 - Fornecedores específicos

```
SELECT codFornecedor, nomeFornecedor
FROM Fornecedor
WHERE codFornecedor IN (1, 3, 5, 7);
```

-- =====

-- CRITÉRIO 8: 3 consultas com funções de agregação (SUM(), AVG(), etc.) (2,0 pts)

-- =====

-- Consulta 8.1 - Soma total de produtos em estoque

```
SELECT SUM(qdtProduto) as Total_Produtos_Estoque
FROM Estoque;
```

-- Consulta 8.2 - Média de compras por cliente

```
SELECT AVG(qdtCompra) as Media_Compras
FROM Cliente;
```

-- Consulta 8.3 - Contagem de produtos por fornecedor

```
SELECT f.nomeFornecedor, COUNT(fo.codProduto) as Total_Produtos
FROM Fornecedor f
JOIN fornece fo ON f.codFornecedor = fo.codFornecedor
GROUP BY f.nomeFornecedor;
```

-- =====

-- CRITÉRIO 9: 2 consultas com funções de datas (NOW(), DATE(), YEAR(), etc.) (3,0 pts)

-- =====

-- *Consulta 9.1 - Data atual e análise temporal*

SELECT

NOW() as Data_Atual,
YEAR(NOW()) as Ano_Atual,
MONTH(NOW()) as Mes_Atual,
DAY(NOW()) as Dia_Atual;

-- *Consulta 9.2 - Análise de período*

SELECT

DATE(NOW()) as Data_Hoje,
DATE_ADD(NOW(), INTERVAL 30 DAY) as Data_30_Dias,
DATE_SUB(NOW(), INTERVAL 7 DAY) as Data_7_Dias_Atras;

-- =====

-- *CRITÉRIO 10: 3 sub-consultas com agrupamento e união de dados (5,0 pts)*

-- =====

-- *Consulta 10.1 - Produtos com estoque acima da média*

SELECT p.nomeProduto, e.qdtProduto
FROM Produto p
JOIN estoque es ON p.codProduto = es.codProduto
JOIN Estoque e ON es.codEstoque = e.codEstoque
WHERE e.qdtProduto > (
 SELECT AVG(qdtProduto)
 FROM Estoque
);

-- *Consulta 10.2 - Fornecedores com mais produtos que a média*

SELECT f.nomeFornecedor, COUNT(fo.codProduto) as Total_Produtos

```

FROM Fornecedor f
JOIN fornece fo ON f.codFornecedor = fo.codFornecedor
GROUP BY f.nomeFornecedor
HAVING COUNT(fo.codProduto) > (
    SELECT AVG(produtos_por_fornecedor)
    FROM (
        SELECT COUNT(codProduto) as produtos_por_fornecedor
        FROM fornece
        GROUP BY codFornecedor
    ) as subquery
);

```

-- Consulta 10.3 - Clientes com compras acima da média

```

SELECT codCliente, qdtCompra
FROM Cliente
WHERE qdtCompra > (
    SELECT AVG(qdtCompra)
    FROM Cliente
);

```

```

-- =====
-- CRITÉRIO 11: 3 consultas com JOIN e visualização de tabelas (6,0 pts)
-- =====

```

-- Consulta 11.1 - Produtos e seus fornecedores

```

SELECT p.nomeProduto, f.nomeFornecedor
FROM Produto p
JOIN fornece fo ON p.codProduto = fo.codProduto
JOIN Fornecedor f ON fo.codFornecedor = f.codFornecedor
ORDER BY p.nomeProduto;

```


-- Consulta 11.2 - Clientes e produtos adquiridos

```
SELECT c.codCliente, c.qdtCompra, p.nomeProduto
FROM Cliente c
JOIN adquire a ON c.codCliente = a.codCliente
JOIN Produto p ON a.codProduto = p.codProduto
ORDER BY c.codCliente;
```

-- Consulta 11.3 - Produtos e estoques

```
SELECT p.nomeProduto, e.codEstoque, e.qdtProduto
FROM Produto p
JOIN estoque es ON p.codProduto = es.codProduto
JOIN Estoque e ON es.codEstoque = e.codEstoque
ORDER BY p.nomeProduto;
```

-- =====

-- CRITÉRIO 12: 4 consultas com tipos de JOIN: INNER, LEFT, RIGHT (6,0 pts)

-- =====

-- Consulta 12.1 - INNER JOIN: Produtos e fornecedores

```
SELECT p.nomeProduto, f.nomeFornecedor
FROM Produto p
INNER JOIN fornece fo ON p.codProduto = fo.codProduto
INNER JOIN Fornecedor f ON fo.codFornecedor = f.codFornecedor
WHERE f.codFornecedor = 1;
```

-- Consulta 12.2 - LEFT JOIN: Todos os produtos e seus fornecedores

```
SELECT p.nomeProduto, f.nomeFornecedor
FROM Produto p
LEFT JOIN fornece fo ON p.codProduto = fo.codProduto
LEFT JOIN Fornecedor f ON fo.codFornecedor = f.codFornecedor;
```

-- Consulta 12.3 - RIGHT JOIN: Todos os fornecedores e seus produtos

```
SELECT f.nomeFornecedor, p.nomeProduto
FROM Produto p
RIGHT JOIN fornece fo ON p.codProduto = fo.codProduto
RIGHT JOIN Fornecedor f ON fo.codFornecedor = f.codFornecedor;
```

-- Consulta 12.4 - INNER JOIN: Clientes e produtos com estoque

```
SELECT c.codCliente, p.nomeProduto, e.qdtProduto
FROM Cliente c
INNER JOIN adquire a ON c.codCliente = a.codCliente
INNER JOIN Produto p ON a.codProduto = p.codProduto
INNER JOIN estoque es ON p.codProduto = es.codProduto
INNER JOIN Estoque e ON es.codEstoque = e.codEstoque
WHERE e.qdtProduto > 100;
```