

ALUNO: Thiago Henrique Gomes Coutinho
Matrícula: 2022211510098

Respostas Lista 03

1. SELECT NumeroDoPedido, NomeDoProduto, Quantidade, PrecoUnitario, (Quantidade * PrecoUnitario) AS Total FROM DetalhesDoPedido WHERE Quantidade < 10;
2. SELECT NomeDoFuncionario, Cargo FROM Funcionarios WHERE Cargo LIKE 'Representante%';
3. SELECT NumeroDoPedido, DataDoPedido, Frete FROM Pedidos WHERE Frete > 100 AND DataDoPedido BETWEEN '1996-07-04' AND '1996-07-16';
4. SELECT NomeDoProduto FROM Produtos WHERE QuantidadePorUnidade LIKE '%bottle%';
5. SELECT CodigoDoCliente, NomeDaEmpresa, Endereco, Cidade, Pais FROM Clientes WHERE Pais IN ('Brasil', 'USA');
6. SELECT CodigoDoProduto, NomeDoProduto FROM Produtos WHERE Descontinuado = 1;
7. SELECT NomeDaCategoria, Descricao FROM Categorias WHERE Descricao LIKE '%sweet%';
8. SELECT CodigoDoFuncionario, Nome FROM Funcionarios WHERE Nome LIKE 'A%';
9. SELECT NomeDoProduto, QuantidadePorUnidade, PrecoUnitario FROM Produtos WHERE NomeDoProduto LIKE '%chocolate%';
10. SELECT NomeDoProduto, NivelDeReposicao FROM Produtos WHERE NivelDeReposicao BETWEEN 10 AND 20;
11. SELECT DISTINCT Cidade FROM Clientes;
12. SELECT NomeDaEmpresa, Telefone FROM Fornecedores WHERE CEP IS NULL;
13. SELECT CodigoDoProduto, NomeDoProduto, UnidadesEmEstoque, UnidadesPedidas FROM Produtos WHERE UnidadesEmEstoque < UnidadesPedidas;
14. SELECT CodigoDoPedido, DataDoPedido FROM Pedidos WHERE DataDoPedido LIKE '1996-08%';
15. SELECT P.NumeroDoPedido, P.Frete FROM Pedidos AS P INNER JOIN Transportadoras AS T ON P.Via = T.CodigoDaTransportadora WHERE T.NomeDaEmpresa = 'Federal Shipping' AND P.Frete >= 100 ORDER BY P.NumeroDoPedido ASC;
16. SELECT MAX(PrecoUnitario) FROM Produtos WHERE NomeDoProduto LIKE '%cheese%';
17. SELECT COUNT(DISTINCT Pais) FROM Fornecedores;
18. SELECT AVG(Frete) FROM Pedidos WHERE DataDoPedido LIKE '1997%';
19. SELECT CodigoDoFuncionario, COUNT(NumeroDoPedido) FROM Pedidos GROUP BY CodigoDoFuncionario;
20. SELECT MAX(Frete) FROM Pedidos WHERE PaisDestinatario = 'Finland';
21. SELECT DataDoPedido, SUM(Total) FROM DetalhesDoPedido GROUP BY DataDoPedido HAVING SUM(Total) > 10000;
22. SELECT NomeDoFuncionario, TotalDeVendas FROM Funcionarios AS F JOIN (SELECT CodigoDoFuncionario, COUNT(*) AS TotalDeVendas FROM Pedidos GROUP BY CodigoDoFuncionario) AS V ON F.CodigoDoFuncionario = V.CodigoDoFuncionario ORDER BY TotalDeVendas DESC;
23. SELECT TOP 5 NomeDoCliente, ValorTotal FROM Clientes AS C JOIN (SELECT CodigoDoCliente, SUM(PrecoUnitario * Quantidade) AS ValorTotal FROM DetalhesDoPedido GROUP BY CodigoDoCliente) AS D ON C.CodigoDoCliente = D.CodigoDoCliente ORDER BY ValorTotal DESC;

24. SELECT NomeDoFuncionario, Cargo, DataDeAdmissao FROM Funcionarios WHERE YEAR(DataDeAdmissao) = 1994;

25. SELECTCodigoDoProduto, NomeDoProduto FROM Produtos WHERE QuantidadePorUnidade LIKE '%12 pk%' AND PrecoUnitario < 20;

26. SELECT Cidade, COUNT(*) AS NumeroDeClientes FROM Clientes GROUP BY Cidade ORDER BY NumeroDeClientes DESC;

27. SELECT CodigoDoFuncionario, NomeDoFuncionario FROM Funcionarios WHERE Regiao = 'WA';

28. SELECT CategoriaID, COUNT(*) AS TotalProdutos FROM Produtos GROUP BY CategoriaID HAVING COUNT(*) > 10;

29. SELECT NomeDoProduto, PrecoUnitario, UnidadesEmEstoque FROM Produtos WHERE UnidadesEmEstoque > 100 ORDER BY PrecoUnitario DESC, UnidadesEmEstoque DESC;

30. SELECT P.CodigoDoProduto, P.NomeDoProduto, SUM(D.Quantidade) AS TotalVendido FROM Produtos AS P JOIN DetalhesDoPedido AS D ON P.CodigoDoProduto = D.CodigoDoProduto GROUP BY P.CodigoDoProduto, P.NomeDoProduto ORDER BY TotalVendido DESC;

31. SELECT NomeDoCliente FROM Clientes WHERE CodigoDoCliente IN (SELECT DISTINCT CodigoDoCliente FROM Pedidos WHERE DataDoPedido BETWEEN '1997-01-01' AND '1997-12-31');

32. SELECT CodigoDoFuncionario, COUNT(*) AS TotalDePedidos FROM Pedidos WHERE DataDoPedido BETWEEN '1996-07-01' AND '1996-09-30' GROUP BY CodigoDoFuncionario;

33. SELECT DISTINCT NomeDoProduto FROM Produtos WHERE CategoriaID IN (SELECT CategoriaID FROM Categorias WHERE Descricao = 'Beverages');

34. SELECT NomeDoCliente, Endereco FROM Clientes WHERE Cidade = 'London' AND Regiao IS NULL;

35. SELECT P.NomeDoProduto, D.Quantidade, D.PrecoUnitario, (D.Quantidade * D.PrecoUnitario) AS Total FROM DetalhesDoPedido AS D INNER JOIN Produtos AS P ON D.CodigoDoProduto = P.CodigoDoProduto WHERE D.NumeroDoPedido = 10250;

36. SELECT E.NomeDoFuncionario, COUNT(P.NumeroDoPedido) AS TotalPedidos FROM Funcionarios AS E INNER JOIN Pedidos AS P ON E.CodigoDoFuncionario = P.CodigoDoFuncionario WHERE P.DataDoPedido >= '1998-01-01' GROUP BY E.NomeDoFuncionario;

37. SELECT NomeDoProduto, PrecoUnitario FROM Produtos WHERE PrecoUnitario > ALL (SELECT PrecoUnitario FROM Produtos WHERE CategoriaID = 2);

38. SELECT TOP 1 NomeDoFuncionario FROM Funcionarios WHERE Cidade = 'Seattle' ORDER BY DataDeAdmissao DESC;

39. SELECT AVG(PrecoUnitario) AS MediaPreco, CategoriaID FROM Produtos GROUP BY CategoriaID HAVING AVG(PrecoUnitario) > 50;

40. SELECT C.NomeDoCliente, COUNT(*) AS NumPedidos FROM Clientes AS C INNER JOIN Pedidos AS P ON C.CodigoDoCliente = P.CodigoDoCliente WHERE P.DataDoPedido BETWEEN '1997-01-01' AND '1997-12-31' GROUP BY C.NomeDoCliente HAVING COUNT(*) >= 3;

41. SELECT C.CodigoDoCliente, C.NomeDoCliente FROM Clientes AS C LEFT JOIN Pedidos AS P ON C.CodigoDoCliente = P.CodigoDoCliente WHERE P.NumeroDoPedido IS NULL;

42. SELECT NomeDoProduto, COUNT(D.NumeroDoPedido) AS QtdPedidos FROM Produtos AS P INNER JOIN DetalhesDoPedido AS D ON P.CodigoDoProduto = D.CodigoDoProduto GROUP BY NomeDoProduto HAVING COUNT(D.NumeroDoPedido) < 10;

43. SELECT P.NomeDoProduto, C.NomeDaCategoria FROM Produtos AS P INNER JOIN Categorias AS C ON P.CategoriaID = C.CategoriaID WHERE P.PrecoUnitario > (SELECT AVG(PrecoUnitario) FROM Produtos WHERE CategoriaID = C.CategoriaID);

44. SELECT F.NomeDoFuncionario, COUNT(P.NumeroDoPedido) AS TotalDePedidos FROM Funcionarios AS F INNER JOIN Pedidos AS P ON F.CodigoDoFuncionario =

P.CodigoDoFuncionario WHERE YEAR(P.DataDoPedido) = 1997 GROUP BY
F.NomeDoFuncionario;

45. SELECT P.NomeDoProduto, I.QuantidadeEmEstoque, I.NivelDeReposicao FROM
Produtos AS P INNER JOIN ItensEmEstoque AS I ON P.CodigoDoProduto = I.CodigoDoProduto
WHERE I.QuantidadeEmEstoque < I.NivelDeReposicao;

46. SELECT DISTINCT P.NomeDoProduto FROM Produtos AS P INNER JOIN
DetalhesDoPedido AS D ON P.CodigoDoProduto = D.CodigoDoProduto WHERE D.Desconto > 0;

47. SELECT F.NomeDoFuncionario, COUNT(P.NumeroDoPedido) AS TotalDePedidos
FROM Funcionarios AS F INNER JOIN Pedidos AS P ON F.CodigoDoFuncionario =
P.CodigoDoFuncionario WHERE P.DataDoPedido LIKE '1996%' GROUP BY F.NomeDoFuncionario
ORDER BY COUNT(P.NumeroDoPedido) DESC;

48. SELECT TOP 5 NomeDoCliente, Cidade FROM Clientes WHERE Regiao = 'WA'
ORDER BY NomeDoCliente;

49. SELECT C.NomeDaCategoria, COUNT(P.CodigoDoProduto) AS
QuantidadeDeProdutos FROM Categorias AS C INNER JOIN Produtos AS P ON C.CategoriaID =
P.CategoriaID GROUP BY C.NomeDaCategoria ORDER BY COUNT(P.CodigoDoProduto) DESC;

50. SELECT E.PrimeiroNome, E.UltimoNome, COUNT(DISTINCT P.CodigoDoCliente)
AS QtdClientesServidos FROM Funcionarios AS E INNER JOIN Pedidos AS P ON
E.CodigoDoFuncionario = P.CodigoDoFuncionario GROUP BY E.PrimeiroNome, E.UltimoNome;

Resposta Lista 04

1. Inserir dois registros na tabela Transportadora:

```
INSERT INTO Transportadora (CodigoDaTransportadora, NomeDaEmpresa, Telefone)
VALUES (4, 'Viaje Bem', '2354556'), (5, 'Transporte Seguro', '2536598');
```

2. Inserir mais duas novas categorias na tabela categorias:

```
INSERT INTO Categorias (CodigoDaCategoria, NomeDaCategoria, Descricao, Figura)
VALUES (9, 'Crustáceo', 'Comida do mar', null), (10, 'Alga', 'Comida das rochas', null);
```

3. Inserir dois novos produtos na tabela produto:

```
INSERT INTO Produto (CodigoDoProduto, NomeDoProduto, Descontinuado)
VALUES (78, 'Arroz', 0), (79, 'Macarrão', 0);
```

4. Inserir um item na tabela Detalhes do Pedido do produto “Genen Shoyu”:

-- Esta instrução supõe que existam as tabelas e colunas apropriadas.

-- O valor para o produto "Genen Shoyu" não foi fornecido; seria necessário o código do
produto correspondente.

```
INSERT INTO DetalhesDoPedido (CodigoDoProduto, Quantidade, NumeroDoPedido,
CodigoDoCliente)
VALUES ((SELECT CodigoDoProduto FROM Produto WHERE NomeDoProduto = 'Genen Shoyu'),
4, 11062, (SELECT CodigoDoCliente FROM Cliente WHERE NomeDoCliente = 'Reggiani
Caseifici'));
```

5. Inserir mais dois novos funcionários (a seu critério):

-- Será necessário preencher com informações adequadas de acordo com a estrutura da tabela Funcionarios.

```
INSERT INTO Funcionarios (Nome, Sobrenome, DataContratacao, etc.)  
VALUES ('Nome1', 'Sobrenome1', 'Data1', etc.), ('Nome2', 'Sobrenome2', 'Data2', etc.);
```

6. Inserir mais uma Transportadora, deixando em branco o campo Telefone:

```
INSERT INTO Transportadora (CodigoDaTransportadora, NomeDaEmpresa)  
VALUES (6, 'Transporte Imaginário');
```

7. Inserir mais um novo fornecedor (a seu critério):

-- Será necessário preencher com informações adequadas de acordo com a estrutura da tabela Fornecedores.

```
INSERT INTO Fornecedores (NomeDaEmpresa, ContatoNome, etc.)  
VALUES ('Fornecedor Novo', 'Contato Novo', etc.);
```

8. Inserir mais duas categorias (a seu critério):

-- Similar ao comando da questão 2, substituindo os valores conforme desejado.

9. Inserir mais dois Fornecedores de acordo com os dados da tabela abaixo:

```
INSERT INTO Fornecedores (CodigoDoFornecedor, NomeDaEmpresa)  
VALUES (30, 'Compre Bem'), (31, 'RC – Compras');
```

10. Inserir mais uma transportadora:

```
INSERT INTO Transportadora (CodigoDaTransportadora, NomeDaEmpresa, Telefone)  
VALUES (6, 'Cometa', '2315647');
```

11. Alterar o campo Tratamento de “Sra.” para “Sr(a).” de todos os registros da tabela funcionários:

```
UPDATE Funcionarios  
SET Tratamento = 'Sr(a).'  
WHERE Tratamento = 'Sra.';
```

12. Alterar o preço do frete de todos os pedidos feitos para a transportadora “United Package”, acrescentando mais R\$ 7,00:

-- Assumindo que o nome da coluna que identifica a transportadora na tabela de pedidos seja 'TransportadoraNome' e o preço do frete 'PrecoDoFrete'.

```
UPDATE Pedidos  
SET PrecoDoFrete = PrecoDoFrete + 7  
WHERE TransportadoraNome = 'United Package';
```

13. Altere os pedidos com frete para a empresa “United Package” relocando para a empresa “Federal Shipping”:

-- Assumindo que a tabela de pedidos tenha uma coluna 'TransportadoraNome' que armazene o nome da transportadora.

```
UPDATE Pedidos
SET TransportadoraNome = 'Federal Shipping'
WHERE TransportadoraNome = 'United Package';
```

14. Alterar todos os telefones nulos da tabela Funcionários para o número “555-5555”:

```
UPDATE Funcionarios
SET Telefone = '555-5555'
WHERE Telefone IS NULL;
```

15. Alterar o salário de todos os funcionários, aumentando em 10%:

```
UPDATE Funcionarios
SET Salario = Salario * 1.1;
```

16. Alterar o nome da categoria “Confections” para “Sweets and Confections”:

```
UPDATE Categorias
SET NomeDaCategoria = 'Sweets and Confections'
WHERE NomeDaCategoria = 'Confections';
```

17. Excluir os registros de funcionários com o sobrenome “Davolio”:

```
DELETE FROM Funcionarios
WHERE Sobrenome = 'Davolio';
```

18. Excluir os registros de clientes que não fizeram pedidos:

```
DELETE FROM Clientes
WHERE CodigoDoCliente NOT IN (SELECT DISTINCT CodigoDoCliente FROM Pedidos);
```

19. Excluir os registros de produtos que estejam descontinuados:

```
DELETE FROM Produto
WHERE Descontinuado = 1;
```

20. Excluir os registros de produtos com preço de custo maior que R\$100,00:

```
DELETE FROM Produto
WHERE PrecoDeCusto > 100.00;
```

21. Excluir todos os pedidos feitos à transportadora “Ship to Shore”:

```
DELETE FROM Pedidos
WHERE TransportadoraNome = 'Ship to Shore';
```

22. Excluir todos os produtos da categoria “Frutos do Mar”:

```
DELETE FROM Produto
WHERE CodigoDaCategoria = (SELECT CodigoDaCategoria FROM Categorias WHERE
NomeDaCategoria = 'Frutos do Mar');
```

23. Selecionar todos os produtos da categoria “Bebidas”:

```
SELECT *
FROM Produto
WHERE CodigoDaCategoria = (SELECT CodigoDaCategoria FROM Categorias WHERE
NomeDaCategoria = 'Bebidas');
```

24. Selecionar o nome e o preço de todos os produtos da categoria “Condimentos”:

```
SELECT NomeDoProduto, PrecoUnitario
FROM Produto
WHERE CodigoDaCategoria = (SELECT CodigoDaCategoria FROM Categorias WHERE
NomeDaCategoria = 'Condimentos');
```

25. Selecionar o nome dos fornecedores que não têm fax:

```
SELECT NomeDaEmpresa
FROM Fornecedores
WHERE Fax IS NULL;
```

26. Selecionar os pedidos feitos no mês de janeiro de 2023:

```
SELECT *
FROM Pedidos
WHERE DataDoPedido >= '2023-01-01' AND DataDoPedido < '2023-02-01';
```

27. Selecionar os pedidos feitos à transportadora “Federal Shipping”:

```
SELECT *
FROM Pedidos
WHERE TransportadoraNome = 'Federal Shipping';
```

28. Selecionar todos os pedidos que contêm o produto “Tofu”:

```
SELECT p.NumeroDoPedido
FROM Pedidos AS p
JOIN DetalhesDoPedido AS dp ON p.NumeroDoPedido = dp.NumeroDoPedido
JOIN Produto AS prod ON dp.CodigoDoProduto = prod.CodigoDoProduto
WHERE prod.NomeDoProduto = 'Tofu';
```

29. Selecionar o nome dos funcionários que fizeram aniversário no mês de julho:

```
SELECT Nome
FROM Funcionarios
WHERE MONTH(DataDeNascimento) = 7;
```

30. Selecionar o nome e a quantidade em estoque dos produtos com mais de 100 unidades em estoque:

```
SELECT NomeDoProduto, UnidadesEmEstoque
```

```
FROM Produto
WHERE UnidadesEmEstoque > 100;
```

Resposta Lista 05

1. Exibir o total de frete pago:

```
SELECT SUM(Frete) AS 'Total Frete Pago' FROM Pedidos;
```

2. Quantos funcionários foram contratados em 1992:

```
SELECT COUNT(*) AS 'Total de Contratações em 1992' FROM Funcionarios WHERE
YEAR(DataDeContratacao) = 1992;
```

3. Quantos fornecedores são da Alemanha:

```
SELECT COUNT(*) AS 'Fornecedores da Alemanha' FROM Fornecedores WHERE Pais =
'Alemanha';
```

4. Qual o total de quantidades vendidas:

```
SELECT SUM(Quantidade) AS 'Total Quantidades Vendidas' FROM DetalhesDoPedido;
```

5. Qual o valor do frete mais caro:

```
SELECT MAX(Frete) AS 'Frete Mais Caro' FROM Pedidos;
```

6. Qual o valor do desconto mais barato:

```
SELECT MIN(Desconto) AS 'Desconto Mais Barato' FROM DetalhesDoPedido WHERE Desconto >
0;
```

7. Quantos produtos são da categoria Condimentos (considerando
CodigoDaCategoria = 2):

```
SELECT COUNT(*) AS 'Produtos da Categoria Condimentos' FROM Produtos WHERE
CodigoDaCategoria = 2;
```

8. Quantas categorias começam com a letra “C”:

```
SELECT COUNT(*) AS 'Categorias com C' FROM Categorias WHERE NomeDaCategoria LIKE 'C%';
```

9. Quantos produtos estão com o estoque abaixo do nível de reposição:

```
SELECT COUNT(*) AS 'Produtos Abaixo do Nível de Reposição' FROM Produtos WHERE
UnidadesEmEstoque < NivelDeReposicao;
```

10. Qual o valor total dos preços unitários em Produtos:

```
SELECT SUM(PrecoUnitario) AS 'Total Preços Unitários' FROM Produtos;
```

11. Quanto já foi vendido (PreçoUnitário * Quantidade) do produto cujo código é 51:

```
SELECT SUM(PrecoUnitario * Quantidade) AS 'Total Vendido Produto 51'
FROM DetalhesDoPedido
WHERECodigoDoProduto = 51;
```

12. Quantos pedidos foram feitos pelo cliente cujo código é “ROMEY” no ano de 1996:

```
SELECT COUNT(*) AS 'Pedidos do Cliente ROMEY em 1996'
FROM Pedidos
WHERECodigoDoCliente = 'ROMEY'
AND YEAR(DataDoPedido) = 1996;
```

13. Quantos funcionários fazem aniversário no mês de julho:

```
SELECT COUNT(*) AS 'Aniversariantes de Julho'
FROM Funcionarios
WHERE MONTH(DataDeNascimento) = 7;
```

14. Quantos produtos começam com a “C” ou “G” ou “N”:

```
SELECT COUNT(*) AS 'Produtos com C/G/N'
FROM Produtos
WHERE NomeDoProduto LIKE 'C%' OR NomeDoProduto LIKE 'G%' OR NomeDoProduto LIKE 'N%';
```

15. Quantos clientes estão localizados na Alemanha, México ou Reino Unido:

```
SELECT COUNT(*) AS 'Clientes em Alemanha/México/Reino Unido'
FROM Clientes
WHERE Pais IN ('Alemanha', 'México', 'Reino Unido');
```

16. Quantos clientes não possuem fax:

```
SELECT COUNT(*) AS 'Clientes Sem Fax'
FROM Clientes
WHERE Fax IS NULL OR Fax = '';
```

17. Qual o total existente em estoque (PreçoUnitario*UnidadesEmEstoque):

```
SELECT SUM(PrecoUnitario * UnidadesEmEstoque) AS 'Valor Total em Estoque'
FROM Produtos;
```

18. Qual o total já vendido (PreçoUnitario*Quantidade):

```
SELECT SUM(PrecoUnitario * Quantidade) AS 'Valor Total Vendido'
FROM DetalhesDoPedido;
```

19. Qual o total vendido (PreçoUnitario*Quantidade) do produto cujo código é 19:


```
SELECT SUM(PrecoUnitario * Quantidade) AS 'Total Vendido Produto 19'  
FROM DetalhesDoPedido  
WHERE CodigoDoProduto = 19;
```

20. Quanto de frete foi pago para os pedidos feitos pelo cliente cujo código é “LILAS”:

```
SELECT SUM(Frete) AS 'Total Frete Cliente LILAS'  
FROM Pedidos  
WHERE CodigoDoCliente = 'LILAS';
```