
Desafio SQL Abrangente: Gestão de E-commerce TECH

História: Plataforma "Compra Fácil"

A plataforma de e-commerce **Compra Fácil** precisa de um sistema robusto para gerenciar seu fluxo de vendas. O sistema deve rastrear o **Cliente**, os **Pedidos** realizados, os **Produtos** comprados, os **Fornecedores** e as **Entregas** feitas pela **Transportadora**.

O objetivo é testar o domínio completo do SQL, desde a estrutura do banco de dados até a extração de métricas complexas de vendas e estoque.

Estrutura das 8 Tabelas (DDL)

Tabela	Colunas (e Tipos Sugeridos)	Chaves e Relacionamentos
1. CLIENTE (CLI)	codCliente (INT), nomeCliente (VARCHAR), email (VARCHAR), dataCadastro (DATE)	PK: codCliente
2. PRODUTO (PRO)	codProduto (INT), nomeProduto (VARCHAR), valorUnitario (DECIMAL), codFornecedor (INT)	PK: codProduto FK: codFornecedor (liga a FORNECEDOR)
3. FORNECEDOR (FOR)	codFornecedor (INT), nomeFornecedor (VARCHAR), cidade (VARCHAR)	PK: codFornecedor

4. PEDIDO (PED)	codPedido (INT), codCliente (INT), dataPedido (DATE), valorTotal (DECIMAL)	PK: codPedido FK: codCliente (liga a CLIENTE)
5. ITEM_PEDIDO (IP)	codPedido (INT), codProduto (INT), quantidade (INT), precoVenda (DECIMAL)	PK Composta: (codPedido, codProduto) FKs: codPedido (liga a PEDIDO), codProduto (liga a PRODUTO)
6. TRANSPORTADORA (TRA)	codTransportadora (INT), nomeTransportadora (VARCHAR), cnpj (VARCHAR)	PK: codTransportadora
7. ENTREGA (ENT)	codEntrega (INT), codPedido (INT), codTransportadora (INT), statusEntrega (VARCHAR), dataEnvio (DATE)	PK: codEntrega FKs: codPedido (liga a PEDIDO), codTransportadora (liga a TRANSPORTADORA)
8. ESTOQUE (EST)	codEstoque (INT), codProduto (INT), quantidadeAtual (INT), localizacao (VARCHAR)	PK: codEstoque FK: codProduto (liga a PRODUTO)

⌚ Atividades Práticas de SQL (30 Exercícios Obrigatórios)

Parte 1: Estrutura e Manipulação (DDL e DML)

- CREATE TABLE & PRIMARY KEY & FOREIGN KEY:** Crie as 8 tabelas acima, definindo todas as PKs e Fks.
- INSERT:** Popule **todas** as 8 tabelas com no mínimo 20 registros cada.

3. **ALTER TABLE & ADD COLUMN:** Adicione a coluna descontoMaximo (DECIMAL(5, 2)) à tabela PRODUTO.
4. **UPDATE & AND:** Aplique um aumento de 10% no valorUnitario de todos os produtos do codFornecedor **igual a 5** E cujo valorUnitario está **abaixo de 100.00**.
5. **DELETE & OR:** Remova da tabela FORNECEDOR todos os fornecedores da '**Cidade A**' OU cujo nomeFornecedor **começa com 'B'**.

Parte 2: Consultas Simples e Filtros (DQL)

6. **SELECT & ALIASES & ORDER BY:** Liste o codCliente e o email, dando um **Alias** de Identificador para o código. Ordene o resultado pela dataCadastro (ASC).
7. **SELECT & BETWEEN:** Liste o nomeCliente e a dataCadastro para clientes que se cadastraram entre '2024-01-01' e '2024-12-31'.
8. **SELECT & LIKE & NOT:** Liste o nomeProduto e o valorUnitario para produtos cujo nome **NÃO contém 'Notebook'** E cujo valorUnitario é **superior a 500.00**.
9. **SELECT & IN:** Mostre o nomeFornecedor e a cidade para fornecedores localizados em ('**São Paulo**', '**Rio de Janeiro**', '**Belo Horizonte**').
10. **SELECT & WHERE & NOT IN:** Liste o nomeTransportadora para transportadoras cujo codTransportadora **NÃO está** na lista **(1, 5, 10, 15)**.

Parte 3: Joins, Filtros Complexos e Agregação (DQL Avançado)

11. **INNER JOIN (3 tabelas) & WHERE:** Liste o **nome do cliente**, o **código do pedido** e o **valor total**. Filtre para pedidos com valorTotal **acima de 500.00**.
 - *Dica: Use CLIENTE -> PEDIDO.*
12. **INNER JOIN (4 tabelas) & ORDER BY:** Liste o **nome do produto**, o **nome do fornecedor** e a **cidade** do fornecedor. Ordene o resultado pelo nome do fornecedor (DESC).
 - *Dica: Use PRODUTO -> FORNECEDOR.*
13. **INNER JOIN (4 tabelas) & AND:** Mostre o codPedido, o nomeProduto e a quantidade vendida. Filtre para pedidos feitos **após '2025-01-01'** E onde a quantidade vendida seja **maior que 2**.
 - *Dica: Use PEDIDO -> ITEM_PEDIDO -> PRODUTO.*
14. **INNER JOIN (5 tabelas) & OR:** Liste o nomeCliente, nomeProduto e dataPedido. Filtre para clientes que moram na '**Cidade B**' (assumindo que a cidade está na tabela CLIENTE por simplificação) OU que compraram o produto com codProduto **igual a 15**.
15. **INNER JOIN (2 tabelas) & GROUP BY & COUNT:** Calcule **quantos produtos** distintos cada codFornecedor fornece. Liste o codFornecedor e a contagem (Alias ProdutosFornecidos).
16. **INNER JOIN (2 tabelas) & GROUP BY & MIN:** Encontre o **menor** valorUnitario de produto para cada cidade de fornecedor. Liste a cidade e o valor mínimo (Alias ProdutoMaisBarato).

17. **INNER JOIN (2 tabelas) & GROUP BY & MAX:** Calcule o **maior** quantidadeAtual em estoque por localizacao. Liste a localizacao e a quantidade máxima (Alias EstoqueMax).
18. **INNER JOIN (3 tabelas) & NOT:** Liste o nomeCliente e o nomeTransportadora. Filtre para pedidos que **NÃO** têm statusEntrega '**Entregue**' E que **NÃO** foram enviados pela transportadora com codTransportadora **igual a 7**.
19. **INNER JOIN (3 tabelas) & LIKE & BETWEEN:** Liste o nomeTransportadora e o statusEntrega. Filtre para transportadoras cujo nomeTransportadora **começa** com 'T' E o codEntrega está **entre 100 e 200**.
20. **INNER JOIN (5 tabelas) & IN & ORDER BY:** Liste o nomeCliente, nomeProduto, dataPedido e statusEntrega. Filtre para produtos com codProduto **IN (2, 4, 6)**. Ordene pela dataEnvio (DESC).

Parte 4: Inclusão de 10 Novos Exercícios (21 a 30)

21. **INNER JOIN (3 tabelas) & GROUP BY & SUM:** Calcule o **total de vendas** (SUM de precoVenda * quantidade) por codPedido. Liste o codPedido e a soma (Alias FaturamentoBruto).
22. **INNER JOIN (2 tabelas) & GROUP BY & AVG:** Calcule o **valor médio** dos pedidos (valorTotal) por codCliente. Liste o codCliente e a média (Alias TicketMedio).
23. **INNER JOIN (4 tabelas) & NOT LIKE:** Liste o nomeCliente, dataPedido e statusEntrega. Filtre para clientes cujo nomeCliente **NÃO contém** 'Junior' OU cujas entregas **NÃO** estão no statusEntrega '**Enviado**'.
24. **INNER JOIN (5 tabelas) & ORDER BY (Múltiplas Colunas):** Liste o nomeProduto, nomeFornecedor e quantidadeAtual em estoque. Ordene primeiro pelo nomeFornecedor (ASC) e depois pela quantidadeAtual (DESC).
25. **INNER JOIN (4 tabelas) & BETWEEN & AND:** Liste o nomeCliente, nomeProduto e precoVenda. Filtre para pedidos realizados **entre '2025-03-01' e '2025-05-31'** E onde o precoVenda está **acima de 200.00**.
26. **INNER JOIN (3 tabelas) & GROUP BY & COUNT (Contagem Distinta):** Calcule quantos **clientes distintos** fizeram pedidos em cada mês (dataPedido). Liste o mês e a contagem (assumindo que você pode extrair o mês).
27. **INNER JOIN (2 tabelas) & UPDATE:** Altere a quantidadeAtual na tabela ESTOQUE para **zero** para todos os produtos que têm codProduto **IN (7, 8, 9, 11)**.
28. **INNER JOIN (3 tabelas) & LIKE & NOT IN:** Liste o nomeCliente e a dataPedido. Filtre para clientes cujo email **contém** '@gmail.com' E cujo codCliente **NÃO está** em **(1, 2, 3)**.
29. **INNER JOIN (4 tabelas) & MIN & GROUP BY:** Calcule o **menor** precoVenda registrado para cada nomeProduto. Liste o produto e o preço mínimo (Alias MenorPrecoHistorico).
30. **INNER JOIN (5 tabelas) & WHERE & OR:** Liste o nomeCliente, nomeProduto, nomeTransportadora e statusEntrega. Filtre para entregas com statusEntrega '**Cancelado**' OU que foram enviadas pela transportadora '**Entrega Rápida**' (assumindo o nome).