

Listagem de Exercícios de MySQL

Este documento contém a estrutura (DDL), população de dados (DML) e uma lista de 60 exercícios de prática em MySQL focados no uso de funções de string, data, numéricas, condicionais, agregações e janelas. A base de dados utilizada é a **lab_functions**.

Estrutura (DDL)

```
-- Criação do banco e tabelas principais
DROP DATABASE IF EXISTS lab_functions;
CREATE DATABASE lab_functions CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;
USE lab_functions;

CREATE TABLE categories (id INT PRIMARY KEY, name VARCHAR(100));
CREATE TABLE products (id INT PRIMARY KEY, name VARCHAR(150), category_id INT, price DECIMAL(10,2), stock INT);
CREATE TABLE customers (id INT PRIMARY KEY, name VARCHAR(150), email VARCHAR(150), city VARCHAR(100));
CREATE TABLE orders (id INT PRIMARY KEY, customer_id INT, order_date DATETIME, status VARCHAR(30), shipped_date DATETIME);
CREATE TABLE order_items (order_id INT, product_id INT, quantity INT, unit_price DECIMAL(10,2), discount DECIMAL(5,2));
CREATE TABLE departments (id INT PRIMARY KEY, name VARCHAR(100));
CREATE TABLE employees (id INT PRIMARY KEY, first_name VARCHAR(100), last_name VARCHAR(100), department_id INT, hire_date DATE, salary DECIMAL(10,2));
CREATE TABLE attendance (id INT PRIMARY KEY, employee_id INT, day DATE, check_in TIME, check_out TIME);
CREATE TABLE payments (id INT PRIMARY KEY, customer_id INT, payment_date DATE, amount DECIMAL(10,2));
CREATE TABLE reviews (id INT PRIMARY KEY, product_id INT, customer_id INT, rating INT, comment VARCHAR(500));
```

População de Dados (DML - resumido)

```
-- Categorias
INSERT INTO categories VALUES (1,'Eletrônicos'),(2,'Livros'),(3,'Casa & Cozinha'),(4,'Esporte'),(5,'Jogos');

-- Produtos
INSERT INTO products VALUES
(101,'Smartphone Delta X',1,1999.90,'2024-11-01 10:00:00'),
(102,'Fone Bluetooth Wave',1,299.00,'2025-01-15 09:30:00'),
(201,'Livro SQL Essencial',2,99.90,'2024-12-10 11:05:00');

-- Clientes
INSERT INTO customers VALUES
(1,'Ana Souza','ana.souza@example.com','Porto Alegre','RS','2024-12-01 09:00:00'),
(2,'Bruno Lima','bruno.lima@example.com','São Paulo','SP','2025-01-10 15:20:00');

-- Pedidos (exemplo)
INSERT INTO orders VALUES (1001,1,'2025-03-15 10:15:00','Shipped','2025-03-16 09:00:00','FastShip',3);
```

Exercícios

- 1 Liste o nome dos produtos em maiúsculas usando UPPER().
- 2 Mostre os 3 primeiros caracteres do nome de cada categoria com SUBSTRING().
- 3 Retorne o nome do produto e seu tamanho (nº de caracteres) com CHAR_LENGTH().
- 4 Exiba o nome dos clientes com espaços removidos nas extremidades usando TRIM().
- 5 Crie uma coluna com o domínio do e-mail usando SUBSTRING_INDEX().
- 6 Concatene first_name e last_name dos funcionários no formato 'Sobrenome, Nome'.
- 7 Substitua a palavra 'Livro' por 'Book' no nome dos produtos com REPLACE().
- 8 Mostre o nome do produto invertido com REVERSE().
- 9 Extraia o nome do estado em minúsculas dos clientes usando LOWER().
- 10 Gere um código curto do produto com as 2 primeiras letras da categoria + id do produto.
- 11 Mostre o nome do cliente e posição do '@' no e-mail com INSTR().
- 12 Padronize city para Title Case usando funções de string.
- 13 Liste pedidos com a data no formato YYYY-MM usando DATE_FORMAT().
- 14 Calcule dias entre order_date e shipped_date com DATEDIFF().
- 15 Exiba o mês por extenso dos pedidos com DATE_FORMAT().
- 16 Some 30 dias a order_date e mostre a data estimada de entrega com DATE_ADD().

- 17 Mostre created_at dos produtos e o dia da semana com DAYNAME().
- 18 Conte pedidos por trimestre usando QUARTER().
- 19 Retorne clientes com antiguidade em meses usando TIMESTAMPDIFF().
- 20 Calcule horas trabalhadas em attendance usando TIMEDIFF().
- 21 Mostre hire_date dos funcionários e anos de casa com TIMESTAMPDIFF().
- 22 Liste pedidos com shipped_date nula exibindo 'Pendente' senão 'Enviado'.
- 23 Mostre a última data do mês de order_date com LAST_DAY().
- 24 Converta order_date para epoch com UNIX_TIMESTAMP().
- 25 Liste produtos com price arredondado para 0, 1 e 2 casas com ROUND().
- 26 Calcule CEIL(price) e FLOOR(price).
- 27 Mostre o valor total do item (quantity * unit_price * (1 - discount)).
- 28 Exiba ABS da diferença entre unit_price e price.
- 29 Aplique aumento condicional no salário usando CASE.
- 30 Calcule média de rating por produto arredondada para 1 casa.
- 31 Gere desconto escalonado fictício com CASE.
- 32 Mostre MOD(quantity, 2) para itens de pedido.
- 33 Crie uma coluna Faixa de Preço baseada em price usando CASE.
- 34 Retorne média, desvio padrão e variância do amount em payments.
- 35 Calcule frete proporcional por item no pedido.
- 36 Exiba número aleatório por produto com RAND() e mostre os 5 maiores.
- 37 Totalize vendas por categoria.
- 38 Conte nº de pedidos por status.
- 39 Mostre ticket médio por cliente.
- 40 Liste os 3 produtos mais vendidos.
- 41 Calcule faturamento por mês.
- 42 Encontre a maior e menor compra por cliente.
- 43 Mostre clientes sem pagamento registrado.
- 44 Calcule % de desconto médio por categoria.
- 45 Retorne notas média, mínima e máxima de reviews por produto.
- 46 Use ROW_NUMBER() para enumerar pedidos por cliente.
- 47 Aplique RANK() para ranquear produtos por faturamento.
- 48 Calcule SUM() acumulado de vendas por mês.
- 49 Use LAG() para mostrar valor do pedido anterior.
- 50 Use LEAD() para prever próximo pagamento.
- 51 Aplique DENSE_RANK() para ranquear salários por departamento.
- 52 Use NTILE(4) para dividir clientes em quartis de gasto.
- 53 Calcule média móvel de 3 meses do faturamento.
- 54 Mostre participação (%) de cada produto no total da categoria.
- 55 Gere um slug de produto (minúsculo, com hífen).
- 56 Monte ID de pedido legível com ORD-AAAA-MM-id.
- 57 Mostre tempo médio entre order_date e shipped_date por shipper.
- 58 Monte relatório de pontualidade dos pedidos (ON_TIME ou LATE).
- 59 Liste funcionários com nome completo, anos de casa e salário reajustado.
- 60 Crie ranking de cidades por valor total comprado e ticket médio.