



Centro Universitário Cidade Verde

Matheus Alende Pires RA: 168649-2024

João Pedro Pereira Marques RA: 180746-2024

João Antonio R. se Souza RA: 152032-2023

PROJETO SEGUNDO BIMESTRE 2025: BANCO DE DADOS FERRAMENTA CERTA.

Maringá
2025

Documentação Técnica - Banco de Dados

Este documento descreve as Funções, Procedures, Triggers e Views do banco de dados, seguindo o formato solicitado.

• 1. FUNÇÕES

• 1.1. ValidarCPF(cpf CHAR(11))

Verifica se um CPF possui 11 dígitos numéricos.

- Parâmetro: cpf - Texto contendo o CPF.
- Retorno: TINYINT(1) (1 para válido, 0 para inválido).
- Uso: Validação de CPF antes de inserir ou atualizar dados de clientes ou funcionários.

• 1.2. CalcularTotalPedido(pid_pedido INT)

Calcula o valor total de um pedido somando o valor dos itens.

- Parâmetro: pid_pedido - ID do pedido.
- Retorno: DECIMAL(10,2) com o valor total do pedido (soma de quantidade * preco_unitario dos itens).
- Uso: Obter o valor total de um pedido específico para faturamento ou relatórios.

• 1.3. CalcularComissaoVendedor(pid_funcionario INT, pmes INT, pano INT)

Calcula a comissão de 3% sobre as vendas pagas de um vendedor em um mês/ano específico.

- Parâmetros: pid_funcionario (ID do vendedor), pmes (Mês), pano (Ano).
- Retorno: DECIMAL(10,2) com o valor da comissão.
- Uso: Calcular a remuneração variável de vendedores com base em seu desempenho mensal.

• 2. PROCEDURES

• 2.1. RealizarVenda(pid_cliente INT, pdata DATE)

Cria um novo pedido na tabela Pedidos com o status inicial "Pendente".

- Parâmetros: pid_cliente (ID do cliente), pdata (Data do pedido).
- Comportamento: Insere um novo registro na tabela Pedidos com os dados fornecidos e status 'Pendente'.

• 2.2. AtualizarEstoque(pid_produto INT, quantidade_comprada INT)

Atualiza (decrementa) a quantidade em estoque de um produto específico.

- Parâmetros: pid_produto (ID do produto), quantidade_comprada (Quantidade a ser subtraída do estoque).
- Comportamento: Executa um UPDATE na tabela Produtos para diminuir o valor da coluna quantidade_estoque.

- **2.3. RelatorioMensalFuncionarios(pmes INT, pano INT)**

Gera um relatório de desempenho mensal dos funcionários ativos.

- Parâmetros: pmes (Mês de referência), pano (Ano de referência).
- Comportamento: Retorna uma consulta com nome, cargo, total de pedidos no mês, valor total de vendas pagas no mês e comissão calculada (usando a função CalcularComissaoVendedor) para cada funcionário ativo, ordenado por vendas decrescentes.

- **3. TRIGGERS**

- **3.1. AtualizarEstoqueAposVenda (AFTER INSERT ON itens_pedido)**

Atualiza automaticamente o estoque de um produto após a inserção de um item em um pedido.

- Evento: Ocorre após a inserção (INSERT) de um registro na tabela itens_pedido.
- Comportamento: Reduz a quantidade_estoque na tabela Produtos com base no id_produto e quantidade do item recém-inserido.

- **3.2. LogAlteracoesCliente (AFTER UPDATE ON clientes)**

Registra um histórico das alterações realizadas nos dados dos clientes.

- Evento: Ocorre após a atualização (UPDATE) de um registro na tabela clientes.
- Comportamento: Insere os dados antigos (OLD.*) do cliente (id, nome, email, cpf, endereco) na tabela Log_Clientes, juntamente com a data e hora da alteração (NOW()).

- **3.3. LogAlteracoesFuncionario (AFTER UPDATE ON funcionarios)**

Registra um histórico das alterações realizadas nos dados dos funcionários.

- Evento: Ocorre após a atualização (UPDATE) de um registro na tabela funcionarios.
- Comportamento: Insere os dados antigos (OLD.*) do funcionário (id, nome, cargo, salario, status) na tabela log_funcionarios, juntamente com a data e hora da alteração (NOW()).

- **4. VIEWS**

- **4.1. view_pedidos_cliente**

Exibe os pedidos juntamente com o nome do cliente correspondente.

- Colunas: cliente (nome do cliente), id_pedido, data_pedido, status.
- Fonte de Dados: Resultado de um JOIN entre as tabelas pedidos e clientes.

- **4.2. view_produtos_mais_vendidos**

Apresenta os produtos ordenados pela quantidade total vendida.

- Colunas: nome (nome do produto), total_vendido (soma das quantidades vendidas).
- Fonte de Dados: Resultado de um JOIN entre itens_pedido e produtos, agrupado por nome do produto.
- Ordenação: Decrescente pela coluna total_vendido.

- **4.3. view_servicos_funcionario**

Mostra detalhes dos serviços agendados ou realizados, incluindo funcionário, cliente e preço.

- Colunas: funcionario (nome), servico (nome), cliente (nome), data_execucao, status, preco.
- Fonte de Dados: Resultado de JOINS entre servicos_funcionarios, funcionarios, servicos e clientes.
- Ordenação: Decrescente pela coluna data_execucao.

- **4.4. view_vendas_funcionario**

Consolida o desempenho de vendas por funcionário ativo.

- Colunas: funcionario (nome), cargo, total_pedidos (contagem total), pedidos_pagos (contagem de pagos), total_vendas (soma do valor dos pedidos pagos).
- Fonte de Dados: Resultado de um LEFT JOIN entre funcionarios e pedidos, filtrado por funcionários ativos e agrupado por funcionário.
- Ordenação: Decrescente pela coluna total_vendas.