

Apostila de Banco de Dados MySQL

Tópicos: Functions, Procedures, Views e Triggers

Esta apostila cobre os principais recursos de programação no MySQL: funções definidas pelo usuário (functions), procedimentos armazenados (procedures), visões (views) e gatilhos (triggers). Além das definições, são apresentados exemplos e scripts para uso em sala de aula.

1. FUNCTIONS (Funções)

Funções são blocos de código armazenados no banco de dados que retornam um valor.

Exemplo de função:

```
CREATE FUNCTION calcular_desconto(valor DECIMAL(10,2), percentual INT)
RETURNS DECIMAL(10,2) RETURN valor - (valor * percentual / 100);
```

Uso:

```
SELECT calcular_desconto(100, 10); -- Resultado: 90.00
```

2. PROCEDURES (Procedimentos Armazenados)

Procedimentos armazenados são rotinas que podem executar várias operações, mas não retornam valor diretamente.

Exemplo de procedure:

```
DELIMITER //
CREATE PROCEDURE listar_clientes()

    SELECT * FROM clientes;
```

Execução: CALL listar_clientes();

3. VIEWS (Visões)

Views são consultas salvas no banco de dados, permitindo reutilização e simplificação de queries.

Exemplo de view:

```
CREATE VIEW vendas_totais AS
SELECT cliente_id, SUM(valor) AS total
FROM vendas
GROUP BY cliente_id;
```

Consulta: SELECT * FROM vendas_totais;

4. TRIGGERS (Gatilhos)

Triggers são blocos automáticos que executam quando certos eventos ocorrem em uma tabela.

Exemplo de trigger:

```
CREATE TRIGGER log_insercao  
AFTER INSERT ON clientes  
FOR EACH ROW
```

```
INSERT INTO log_atividades(descricao, data)  
VALUES (CONCAT('Novo cliente: ', NEW.nome), NOW());
```

5. Estrutura de Tabelas para Demonstração

```
CREATE TABLE `funcionarios` (  
  
  `id_funcionario` int(11) NOT NULL,  
  
  `nome` varchar(60) NOT NULL,  
  
  `salario` float NOT NULL DEFAULT 0  
  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

```
INSERT INTO `funcionarios` (`id_funcionario`, `nome`, `salario`) VALUES  
  
(1, 'José Ferreira da Silva', 3500),  
  
(2, 'João Saldanha', 5000),  
  
(3, 'Maria da Silva', 3500);
```

```
ALTER TABLE `funcionarios`  
  
  ADD PRIMARY KEY (`id_funcionario`);  
  
ALTER TABLE `funcionarios`  
  
  MODIFY `id_funcionario` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;
```

6. Exercícios Propostos

1. Crie uma função que calcule um acréscimo de 10% sobre um valor.
2. Crie uma procedure que insira um novo cliente no banco de dados.
3. Crie uma view que mostre o nome do cliente e o total de suas compras.
4. Crie uma trigger que registre no log toda nova venda inserida.