

Exercícios guiados com function:

1) Crie a base de dados confecção e as tabelas produto e movimentação:

```
CREATE DATABASE confeccao;
USE confeccao;
CREATE TABLE produto( id int auto_increment primary key,
                        nome varchar(30),
                        preco decimal(10,2)
)engine=INNODB default charset=utf8 collate=utf8_unicode_ci;

CREATE TABLE movimentacao( id int auto_increment primary key,
                             produto_id int,
                             quantidade int,
                             operacao varchar(15),
CONSTRAINT fk_movimentacao__produto_id FOREIGN KEY(produto_id)
REFERENCES produto(id) ON DELETE RESTRICT ON UPDATE CASCADE
)engine=INNODB default charset=utf8 collate=utf8_unicode_ci;
```

2) Insira algumas informações nas tabelas:

```
INSERT INTO produto VALUES(1, 'TANGA', 3.50);

INSERT INTO movimentacao(produto_id, quantidade, operacao)
VALUES(1, 10, 'ENTRADA'), (1, 20, 'SAÍDA'), (1, 30, 'ENTRADA');
```

3) Crie uma função chamada estoque. Esta função deve retornar o estoque real do produto informado, tomando por base seu histórico de movimentações:

```
DROP FUNCTION IF EXISTS estoque;
DELIMITER $$;
CREATE FUNCTION estoque(produto int) returns int
BEGIN
    DECLARE entrada, saida, total int;
    SET entrada = (SELECT SUM(quantidade) FROM movimentacao
                   WHERE produto_id=produto AND operacao='ENTRADA');
    SET saida = (SELECT SUM(quantidade) FROM movimentacao
                WHERE produto_id=produto AND operacao='SAÍDA');
    SET total = entrada - saida;
    RETURN total;
END $$
DELIMITER ;
```

4) Construa uma consulta que retorne o nome e o estoque de todos os produtos:

```
SELECT nome, estoque(id) FROM produto;
```

Nesta apostila abordamos um mesmo tipo de função / stored procedure, escrita de maneiras diferentes, utilizando recursos diferentes.