

Faça um procedimento, considerando o modelo de ‘cliente’, ‘pedido’, ‘produto’ e ‘movimento’, de forma que:

- a) Não permita excluir um produto que já tenha sido pedido;
- b) Ao inserir um novo produto o campo ‘nome’ precisa de valor;

OBS.: Defina a quantidade de parâmetros necessários para tratar estas regras.

a)

```
CREATE OR REPLACE FUNCTION
fn_deletar_produto(_NOME char (30)) RETURNS VOID AS $$

DECLARE
    _EXISTE char (30);
BEGIN
    _EXISTE = (  select nome from produto
                  where lower(nome) = lower(_NOME)      );
    IF _EXISTE IS NOT NULL THEN
        PERFORM * from pedido, movimento, produto
        where pedido.nro_ped = movimento.nro_ped
        and movimento.cod_prod = produto.cod_prod
        and lower(produto.nome) = lower(_NOME);
        IF FOUND THEN
            RAISE NOTICE 'Não foi possível deletar pois o produto: "%" possui
            pedido(s.)', upper(_NOME);
        ELSE
            delete from produto
            where lower(produto.nome) = lower(_NOME);
            RAISE NOTICE 'Produto "%" deletado com sucesso!', upper(_NOME) ;
        END IF;
    ELSE
        RAISE EXCEPTION 'Produto "%" não encontrado!', upper(_NOME);
    END IF;
    RETURN;
END;
$$
LANGUAGE PLpgsql;
```

b)

```
CREATE OR REPLACE FUNCTION
fn_insere_produto(_COD int, _NOME char (30), _PRECO float, _CAT char (20))
RETURNS VOID AS $$

BEGIN

IF _NOME != " THEN

    insert into produto values (_COD, _NOME, _PRECO, _CAT);
    RAISE NOTICE 'Produto inserido com sucesso!';

ELSE

    RAISE EXCEPTION 'NOME não preenchido!';

END IF;

RETURN;
END;
$$
LANGUAGE PLpgsql;
```