

Lista de exercícios

- 1) Crie um Procedimento armazenado em PostgreSQL (somente SQL) que insere uma nova linha em uma tabela, composta por três colunas: nome, sobrenome e idade:

Para isto, crie a tabela de teste:

```
CREATE TABLE pessoas (  
    nome varchar(20),  
    sobrenome varchar(40),  
    idade smallint  
);
```

- 2) Criar um procedimento armazenado que permita inserir um novo produto na tabela de produtos de nosso banco de dados.

```
Create TABLE produto (  
    p_cod_produto INT,  
    p_nome_produto VARCHAR(30),  
    p_descricao TEXT,  
    p_preco NUMERIC,  
    p_qtde_estoque SMALLINT  
)
```

- 3) Crie um procedimento que atualize o preço de um produto específico, identificado pelo seu código de produto:
- 4) Crie um procedimento que retorne todos os produtos com quantidade em estoque abaixo de um valor especificado na chamada da procedure.
- 5) Criar um procedimento armazenado que insira um novo produto na tabela de produtos, ao mesmo tempo em que aplica um desconto percentual especificado a todos os produtos logo após.

6) O PostgreSQL permite criar funções para facilitar operações diárias e abstrair a complexidade na leitura e utilização dos códigos. Isto posto, analise a função a seguir:

7) Criar uma função que retorna o número total de registros na tabela COMPANY.

A tabela COMPANY, que tem os seguintes registros –

```
select * from COMPANY;
```

```
id | name | age | address | salary
```

```
----+-----+-----+-----+-----
```

```
1 | Paul | 32 | California| 20000
```

```
2 | Allen | 25 | Texas | 15000
```

```
3 | Teddy | 23 | Norway | 20000
```

```
4 | Mark | 25 | Rich-Mond | 65000
```

```
5 | David | 27 | Texas | 85000
```

```
6 | Kim | 22 | South-Hall| 45000
```

```
7 | James | 24 | Houston | 10000
```

```
(7 rows)
```

8) Quais são as diferentes formas de fazer referencias a parâmetros de uma função?

9) Criar uma função que retorne todos os funcionários que recebem acima de R\$ 3000. Usar laço de repetição.

```
CREATE TABLE empregados(
```

```
codigo serial,
```

```
nome_emp text,
```

```
salario int,
```

```
departamento_cod int,
```

```
PRIMARY KEY (codigo),
```

```
FOREIGN KEY (departamento_cod) REFERENCES departamentos (id));
```

```
CREATE TABLE departamentos (id serial primary key, descricao varchar);
```