1- A Stored Procedures trazem melhorias na performance, já que terá uma menor quantidade de trânsito entre redes e são consideradas altamente portáveis porém um banco de dado pode se tornar altamente dependentes das Stored Procedures e se por algum acaso a base terá que ser reescrita, será necessário reescrever todas as

stored procedures do banco.

3- Normalmente o uso do SQL requer a aplicação cliente deve enviar cada consulta ao servidor de banco de dados, esperar ser processada, receber e processar os resultados, fazer alguns cálculos e, em seguida, enviar outra consulta ao servidor. Com plpgsql é possível agrupar um bloco de computação e uma série de consultas *dentro* do servidor de banco de dados, tendo assim o poder de uma linguagem procedural e a facilidade de uso do SQL, mas com uma economia considerável de sobrecarga de comunicação cliente/servidor:  
  
4) Linha 1: Esta é a cláusula que define uma procedure. A parte "OR REPLACE" indica que, se já existir uma procedure com o nome faz\_algo, ela será substituída.

Linha 2: Indica o início do corpo da procedure. O código PL/pgSQL que define o que a procedure faz vem entre os delimitadores $$.

Linha 3 e 4:Declare é usada para declarar variaveis.

Subtotal é uma variável declarada que servirá como um alias (um nome alternativo) para o primeiro parâmetro de entrada da procedure, que é o valor de x.

**$1**: Refere-se ao primeiro parâmetro da procedure. Como x é o primeiro parâmetro de entrada, o valor de x é referenciado por $1.

Isso permite referenciar x usando o nome subtotal dentro do código da procedure, em vez de usar diretamente $1.

Linha 5, 6 e 7: Begin define o começo da logica do procedure

subtotal + subtotal \* 0.05: Calcula 5% a mais do valor de subtotal (ou seja, x), somando esse valor ao subtotal original.

End define onde o procedure acaba.

Linha 8. Define a linguagem usada para ser plpgsql  
  
7)Linha 1: cria a procedure nome função que recebe informação numeric e retorna varchar com maximo de 20 caracteres e o “as $$” define que é plpgsql.

Linha 2 a 4:”Declare” declara as variaveis “registro” como o tipo record que armazena uma linha de uma tabela e “x alias for $1” referese o primeiro parâmetro da função sendo atribuído ao x

Linha 5 a 10: Begin começa o procedimento do codigo.

FOR registro IN SELECT \* FROM Empregado: Esse é um loop que itera sobre os resultados da consulta SQL.

WHERE salario >= x: O loop seleciona todos os registros da tabela Empregado onde o salário é maior ou igual ao valor de x (o valor passado como parâmetro).

O resultado de cada linha selecionada será armazenado na variável registro

Loop return next marca o inicio do codigo que marca o início do bloco de código que será repetido para cada linha selecionada pela consulta SQL.

RETURN NEXT. registro.nome: Retorna o próximo valor para a função. que é o nome do funcionario.

A função simplesmente repassa uma lista de funcionarios que possuem um salario superior ao que foi inserido no select função. Se eu definir a função com um valor de 2000, apenas funcionarios com um salario maior de 2000 ira ser retornados