

1. (2.0) Construa as instruções DDL necessárias para construir um B.D capaz de armazenar as informações cadastrais dos funcionários de uma empresa.

Dados Cadastrais do Funcionário		
CPF:	Nome:	
Data de Nasc.:	Nacionalidade:	Sexo:
Est. Civil:	RG:	Data de Admissão:
Endereço:	Telefone:	
Cargos Ocupados (Ocupação)		
Cargo:	Dt. Início:	Dt. Fim:
Cargo:	Dt. Início:	Dt. Fim:
Dependentes		
Nome:	Dt. Nascimento:	
Nome:	Dt. Nascimento:	

**Observações:**

- cpf do funcionário é um campo único;
  - endereço é o campo textual atômico;
  - Leve em consideração que o funcionário pode ter vários cargos (ocupações) e dependentes.
  - No próprio B.D deve ser possível consultar e armazenar os nomes de todos os cargos possíveis que um funcionário pode ter dentro da empresa;
  - um dependente é dependente de um único funcionário;
2. Crie para o B.D:
- (0.5) 1 *superuser*: **fulano** com senha **fulano**;
  - (0.5) 1 usuário que pode somente consultar: **ciclano** com senha **ciclano**;
3. (1.0) Construa um STORE PROCEDURE de validação para a coluna *cpf*.

**Dicas:**

- A função desenvolvida nas aulas pode ser usada. Se não possuir a função em mãos, será preciso implementá-la novamente.
- Lembrando que 00000000000, 11111111111, 22222222222, 33333333333, 44444444444, 55555555555, 66666666666, 77777777777, 88888888888 e 99999999999 também são cpf's **INVÁLIDOS**;

# Algoritmo para Validar CPF

Quando se está trabalhando em um sistema corporativo, é comum a necessidade de validar CPF. Muita gente não sabe que um CPF para ser válido não basta apenas atender à máscara “###.###.###-##” (o caractere ‘#’ representa um número), existe uma regra matemática que também deve ser verificada para um CPF ser considerado válido. Se você acha que é complicado verificar se um CPF é válido ou não, você vai se surpreender!

## REGRA PARA VALIDAR CPF

O cálculo para validar um CPF é especificado pelo Ministério da Fazenda, que disponibiliza no próprio site as funções (em javascript) para validação de CPF. Vamos entender como funciona.

O CPF é formado por 11 dígitos numéricos que seguem a máscara “###.###.###-##”, a verificação do CPF acontece utilizando os 9 primeiros dígitos e, com um cálculo simples, verificando se o resultado corresponde aos dois últimos dígitos (depois do sinal “-”).

*Vamos usar como exemplo, um CPF fictício “529.982.247-25”.*

### Validação do primeiro dígito

Primeiramente multiplica-se os 9 primeiros dígitos pela sequência decrescente de números de 10 à 2 e soma os resultados. Assim:

$$5 * 10 + 2 * 9 + 9 * 8 + 9 * 7 + 8 * 6 + 2 * 5 + 2 * 4 + 4 * 3 + 7 * 2$$

O resultado do nosso exemplo é:

**295**

O próximo passo da verificação também é simples, basta multiplicarmos esse resultado por 10 e dividirmos por 11.

$$295 * 10 / 11$$

O resultado que nos interessa na verdade é o RESTO da divisão. Se ele for igual ao **primeiro dígito verificador** (primeiro dígito depois do ‘-’), a primeira parte da validação está correta. **Observação Importante:** Se o resto da divisão for igual a 10, nós o consideramos como 0.

Vamos conferir o primeiro dígito verificador do nosso exemplo:

*O resultado da divisão acima é '268' e o RESTO é 2*

Isso significa que o nosso CPF exemplo passou na validação do primeiro dígito.

### Validação do segundo dígito

A validação do segundo dígito é semelhante à primeira, porém vamos considerar os 9 primeiros dígitos, mais o primeiro dígito verificador, e vamos multiplicar esses 10 números pela sequência decrescente de 11 a 2. Vejamos:

$$5 * 11 + 2 * 10 + 9 * 9 + 9 * 8 + 8 * 7 + 2 * 6 + 2 * 5 + 4 * 4 + 7 * 3 + 2 * 2$$

O resultado é:

**347**

Seguindo o mesmo processo da primeira verificação, multiplicamos por 10 e dividimos por 11.

$$347 * 10 / 11$$

Verificando o RESTO, como fizemos anteriormente, temos:

*O resultado da divisão é '315' e o RESTO é 5*

Verificamos, se o resto corresponde ao segundo dígito verificador.

Com essa verificação, constatamos que o CPF 529.982.247-25 é válido.