

Design de código

Júlia está criando um código para a empresa de sua mãe e ela precisa cadastrar os funcionários da empresa. Com isso ela criou o seguinte código:

```
class Funcionario {
  constructor(nome, cpf, salario) {
    this.nome = nome;
    this.cpf = cpf;
    this._salario = salario;
    this._bonificacao;
  }
}

class Gerente extends Funcionario {
  constructor(nome, cpf) {
    super(nome, cpf, 2400);
    this._bonificacao = 1.2;
  }
}

class Assistente extends Funcionario {
  constructor(nome, cpf) {
    super(nome, cpf, 1200);
    this._bonificacao = 0.9;
  }
}
```

Com essas classes ela tem uma maneira de saber quem são os funcionários dentro da empresa dos seus pais. Porém existem alguns cargos dentro da empresa que só existe um funcionário preenchendo eles. Por isso Júlia decidiu não criar classes filhas e usar diretamente a classe `Funcionário` para identificar esses cargos.

Qual o problema com essa decisão?

Selecione 2 alternativas

- A** Ela terá um grande trabalho no futuro quando quiser adicionar uma classe para um tipo de funcionário que atualmente está sendo representado pela classe `funcionário`.
- B** O problema é que as classes `Gerente` e `Assistente` são específicas demais e não possuem real necessidade. Ela poderia usar apenas a classe `Funcionário`.
- C** Um dos problemas é que ela perde a informação de qual tipo de funcionário ela tem. Como ela está usando uma classe mais genérica ao invés de criar classes específicas ela não distingue cada tipo de funcionário.

