

Herdando de funcionário

Caso você precise do projeto com todas as alterações realizadas na aula passada, você pode baixá-lo por meio [deste link](https://github.com/alura-cursos/kotlin-oo-heranca-polimorfismo-interfaces/archive/aula-1.zip) (<https://github.com/alura-cursos/kotlin-oo-heranca-polimorfismo-interfaces/archive/aula-1.zip>).

Faça com que a classe `Gerente` herde de `Funcionario`, para que consiga reutilizar os membros de funcionário.

Para isso, modifique a assinatura de `Funcionario`, para que utilize a palavra chave `open` que permite outras classes herdarem dela. Em seguida, faça a herança enviando os argumentos necessários, conforme exemplo abaixo:

```
class Gerente(  
    nome: String,  
    cpf: String,  
    salario: Double,  
) : Funcionario(  
    nome = nome,  
    cpf = cpf,  
    salario = salario  
)
```

Então, comente o método `bonificacao()` para evitar o problema de compilação.

Não se preocupe que veremos como lidar com esse problema adequadamente no decorrer do curso.

Rode o código com esses ajuste e veja se funciona. Se tudo certo, crie um novo funcionário para representar um diretor.

Na sequência, faça a mesma implementação feita no gerente, com a diferença que o diretor recebe PLR via construtor.

Após implementar o código, crie um diretor enviando e imprimindo suas informações (properties). Por fim, chame o código de autenticação para o diretor e rode o programa conferindo se apresenta o resultado esperado.