

## Atividade de PROG3

Com base no livro de referência da disciplina:

HORSTMANN, C. S.; CORNELL, G., Core Java Volume I - Fundamentos. 8ª edição. São Paulo: Pearson, 2009. (Ou versão mais recente). Códigos de exemplos em <https://horstmann.com/>

Obs.: Atividade inspirada nos assuntos tratados nos Capítulos 4 (Objetos e Classes) e 5 (Herança).

### > Descrição geral da atividade:

Considere a Figura 1 (na página final) a visão detalhada de classe “Empregado” para uma empresa ABC. Logo, todo colaborador quando contratado são registradas as informações (atributos) básicas: Nome, Salário-base e Data da Contratação. O plano de carreira da empresa prevê também tipos especiais de empregados, tais como:

- Gerente (geral) – fazendo jus a um bônus salarial que é adicionado ao Salário-base;
- Gerente executivo – possui os mesmos benefícios de Gerente geral e agrega um bônus de coparticipação salarial;
- Secretário – trabalha em turno específico: 1- manhã, 2- tarde, ou 3- noite;
- Programador – registra-se textualmente a habilidade de área de programação em atuação.

Assim, observando novamente a Figura 1, atente para:

- 1) Ao criar a superclasse “Empregado”, esta deverá ter os atributos gerais para nome, salário e data da contratação; e, métodos construtor (mesmo nome da classe), “gets” para cada atributo, e um método para aumento salarial (“aumentoSalarial”) que atualizará o atributo salário do empregado com base em um percentual desejado.
- 2) Criar as subclasses de herança (estendidas) à superclasse “Empregado”: “Gerente”, “Secretario” e “Programador”. Observar e avaliar as especificidades de cada subclasse, em termos de atributos, métodos personalizados (com a possibilidade de polimorfismo aos métodos de superclasse).
- 3) “Gerente” – possui atributo de bônus salarial, bem como método “set” para aplicação de valor real de bônus a ser adicionado ao salário, exigindo personalização de método “getSalario”, a considerar a adição do bônus ao salário-base. O método construtor “Gerente” deverá inicializar bônus nulo, posto que tal privilégio possa ser aplicado a qualquer momento ao longo do tempo de trabalho. Veja que gerente “Executivo” agrega ainda um bônus de coparticipação ao salário final.
- 4) “Secretário” e “Programador” também possuem atributos e métodos específicos, conforme vista no diagrama de classes.

### Faça:

- Classe principal (main) .java deverá concentrar os elementos de front-end (GUI para com o usuário), usando o GUI Designer disponível. Logo, esta tela principal deverá invocar as funcionalidades desejadas (que poderão gerar arquivos .java de interface específicos): A) cadastrar e consultar empregados, mantendo listas adequadas (ArrayLists, p. ex.) de cada tipo de empregado: 1-Empregado geral; 2-Gerente geral; 3-Gerente executivo; 4-Secretário; 5-Programador. Observe que no cadastro, a interface GUI deverá

ter elementos necessários para o devido cadastro de um novo empregado, com telas de formulários adequadas. B) Na consulta, deverá apresentar tela organizando tabelas de informações cadastradas para cada categoria de empregado, listando-os. Faça a previsão de um botão “Remover empregado” a fim de permitir seleção de um determinado funcionário e exclusão da lista, atualizando as tabelas de consultas gerais.

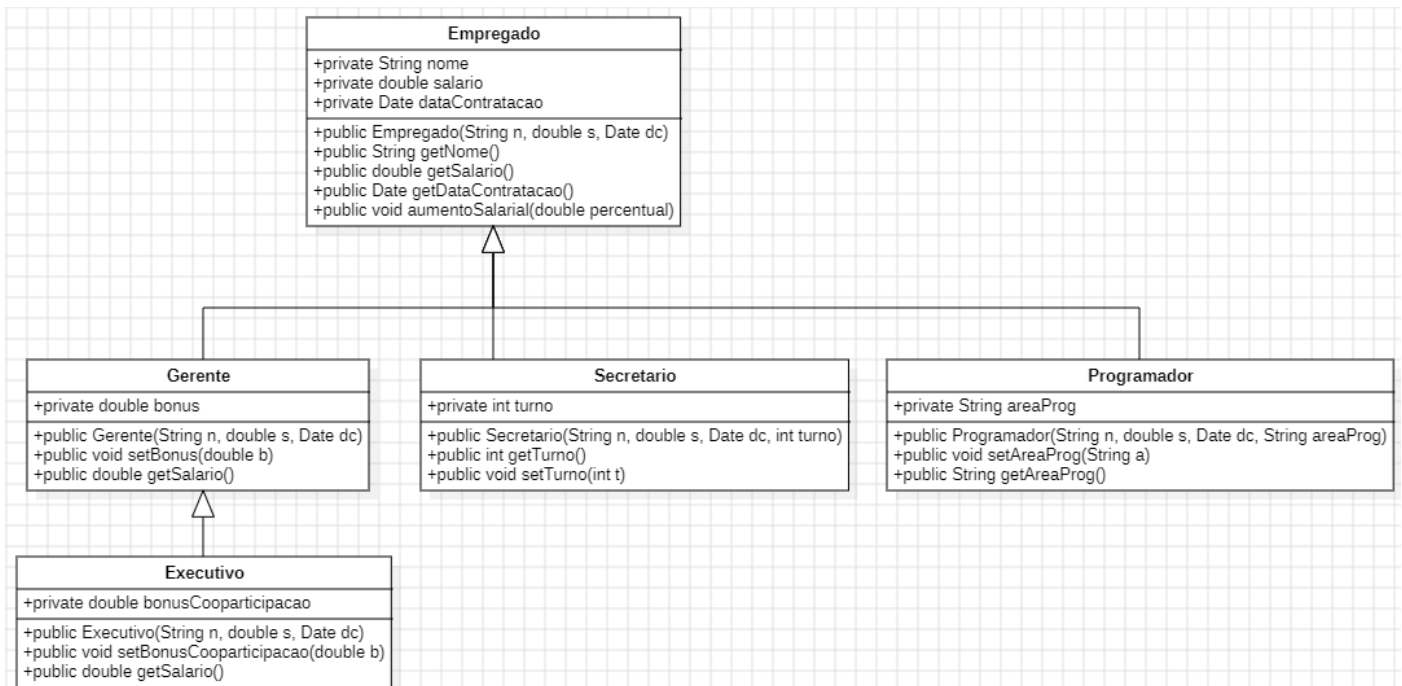


Figura 1 – Diagrama de Classes detalhando a superclasse “Empregado” e generalizações pertinentes.