

Atividade de Programação em Java

Objetivo:

Desenvolver um sistema de gerenciamento de funcionários com as classes **Funcionario** e **Gerente**, aplicando os conceitos de encapsulamento, herança e polimorfismo.

Requisitos:

Classe Funcionario:

- Deve conter atributos privados para **nome** (String) e **salario** (double).
- Fornecer métodos públicos para acesso e modificação desses atributos.
- Implementar um método **exibirInfo()** que mostra as informações do funcionário.

Classe Gerente:

- Deve herdar da classe **Funcionario**.
- Adicionar um atributo **departamento** (String).
- Fornecer métodos públicos para acesso e modificação do atributo **departamento**.
- Sobrescrever o método **exibirInfo()** para incluir informações específicas de gerente.

Teste do Sistema:

- Criar instâncias de **Funcionario** e **Gerente** e demonstrar o uso dos métodos para acessar e modificar atributos.
- Utilizar polimorfismo para chamar o método **exibirInfo()** em ambas as classes.

Desafio Opcional:

- Criar uma interface chamada **Trabalhador** que contenha um método **calcularSalario()**.
- Implementar a interface **Trabalhador** nas classes **Funcionario** e **Gerente**.
- No método **calcularSalario()**, considerar um bônus de 10% para os gerentes.

Critérios de Avaliação:

- Correta implementação de encapsulamento nas classes.
- Uso apropriado de herança e sobrescrita de métodos.
- Demonstração adequada do conceito de polimorfismo.
- Implementação correta e clareza do código.
- Utilização de boas práticas de programação.

Observações:

- Utilize comentários no código para explicar partes importantes.
- Submeta o código-fonte e um documento explicativo.

Desenvolver um sistema de gerenciamento de funcionários com as classes **Funcionario** e **Gerente**, aplicando os conceitos de encapsulamento, herança e polimorfismo.

Requisitos:

Classe Funcionario:

- Deve conter atributos privados para **nome** (String) e **salario** (double).
- Fornecer métodos públicos para acesso e modificação desses atributos.
- Implementar um método **exibirInfo()** que mostra as informações do funcionário.

Classe Gerente:

- Deve herdar da classe **Funcionario**.
- Adicionar um atributo **departamento** (String).
- Fornecer métodos públicos para acesso e modificação do atributo **departamento**.
- Sobrescrever o método **exibirInfo()** para incluir informações específicas de gerente.

Teste do Sistema:

- Criar instâncias de **Funcionario** e **Gerente** e demonstrar o uso dos métodos para acessar e modificar atributos.
- Utilizar polimorfismo para chamar o método **exibirInfo()** em ambas as classes.

Desafio Opcional:

- Criar uma interface chamada `Trabalhador` que contenha um método `calcularSalario()`.
- Implementar a interface `Trabalhador` nas classes `Funcionario` e `Gerente`.
- No método `calcularSalario()`, considerar um bônus de 10% para os gerentes.

Critérios de Avaliação:

- Correta implementação de encapsulamento nas classes.
- Uso apropriado de herança e sobrescrita de métodos.
- Demonstração adequada do conceito de polimorfismo.
- Implementação correta e clareza do código.
- Utilização de boas práticas de programação.

Observações:

- Utilize comentários no código para explicar partes importantes.
- Submeta o código-fonte e um documento explicativo.