Exercícios - Interfaces

1 Crie a interface Funcionario com o método calcularSalario( ) cujo retorno seja *double*. Crie a classe concreta *Operario* que implementa Funcionario. A classe tem os atributos salarioDia e diasTrabalhados. O método calcularSalario( ) retorna a multiplicação de salarioDia por diasTrabalhados. Crie a classe *Gerente* que implementa Funcionario. A classe tem o atributo salarioMes. O método calcularSalario( ) retorna o salarioMes mais um bônus de 10%. Crie a classe FolhaPagamento com uma lista do tipo Funcionario com alguns objetos do tipo *Operario* e alguns do tipo *Gerente*.

a) Mostre a execução polimórfica de calcularSalario( ) utilizando os objetos desta lista.
b) Ordene os objetos pelo salário, do menor para o maior, utilizando o método sort( )

https://github.com/merlinuenp/prog_II_2024/tree/master/src/main/java/capitulo_8/folha

2 No contexto da herança, é possível usar uma variável de referência da superclasse para instanciar um objeto da subclasse. No entanto, a partir desta variável, não é possível chamar os métodos da subclasse.

a) Como fazer para acessar estes métodos da subclasse, se necessário. Por meio de Casting
b) Escreva um exemplo de código em Java que mostre esta situação

3 Pesquise na literatura porque alguns autores dizem ser uma péssima ideia usar herança de classe e sugerem que é melhor usar herança de interface.

Por causa da duplicação de classe.

4 Crie uma interface IPersonagem com o método getPosicao( ). Crie a classe Heroi e a classe Inimigo, ambas implementando a interface. Crie a classe Jogo com um Heroi e um Inimigo. Na classe Jogo deve haver o método colidiu( ) que obtém a posição do do Heroi e a posição do Inimigo; se forem iguais retorna true, se forem diferentes, retorna false.

5 Suponha que, em um determinado contexto, seja necessário criar classes para representar bebidas alcoólicas. Cada bebida deve saber dizer seu teor alcoólico. Mostre a diferença entre fazer isto utilizando classe abstrata e utilizando interface.

A classe pode manter informações de estado com o uso de variáveis de instancia, a interface não,

6 Suponha que você vai criar um novo algoritmo de busca. O algoritmo deve receber uma coleção de objetos e um objeto a ser buscado. Se o objeto estiver na coleção, o retorno é true; se não estiver, o retorno é false. Por exemplo, considere uma lista de arapongas e um objeto *Araponga*:

```
public Boolean buscar( Araponga[ ] lista, Araponga araponga ) { … }
```

public Boolean buscar(IPassaro[] lista, IPassaro araponga){...}

Na forma como está, o método só consegue realizar buscas em coleções do tipo *Araponga*. Modifique o método para que ele receba referência da interface IPassaro, em vez de *Araponga*. Discuta as vantagens e desvantagens dessa alteração.

Faz com que seja possível buscar em todas as instancias que implementam IPassaro, ao invés de apenas na subclasse Araponga

7 Explique como manter em um programa duas ou mais classes com mesmo nome.

Se estiverem em pacotes diferentes

8 Explique por que os métodos definidos em uma interface são implicitamente public.

Porque podem ser utilizados em qualquer código, sendo definido apenas o que uma classe pode fazer, mas não como,, utilizando os métodos especificados na interface.

9 Explique por que uma interface não pode ter variáveis de instância (atributos).

Pois as variáveis são implicitamente public, final e static, e devem ser inicializadas,sendo basicamente constantes.

10 Explique o que é um atributo final. Qual a convenção Java para nome de atributos final?

É um atributo que não pode ser alterado, geralmente escrito em letras maiúsculas.

11 Que pacote Java é importado automaticamente em um programa?

Java.lang, que contém varias classes de uso geral.