



Lista de Exercícios 09 – Herança

Instruções

- *Todos os exercícios que envolvem programas devem ser resolvidos por programas em linguagem Java;*
- *Na solução dos exercícios, devem ser utilizados os conceitos listados no cabeçalho desta lista;*
- *Eventuais dúvidas podem ser sanadas com o professor.*

1. Crie uma classe *Vetor*, em que o único atributo é um vetor de *Strings*. Nesta classe, sobrescreva os métodos *clone()* e *equals()* da classe *Object*. Demonstre a diferença entre cópia rasa e cópia profunda de objetos desta classe através da utilização do métodos *clone()* (versões original e sobrescrita) e da comparação dos resultados através do método *equals()* e do operador *==*.
2. Escreva Uma classe *Pessoa* com atributos:
 - nome (tipo *String*);
 - sobrenome (tipo *String*).

A classe ainda deve conter os seguintes métodos:

- a. *getters* e *setters*;
- b. *getNomeCompleto()* que não possui parâmetros de entrada e que retorna a concatenação do atributo nome com o atributo sobrenome (utilize *toString()*);
- c. Um construtor sem parâmetros;
- d. Outro construtor que recebe como parâmetros o nome e o sobrenome da pessoa.

Implemente também uma subclasse de *Pessoa*, chamada *Funcionario*. A classe *Funcionario* deve ter os atributos *matricula* (tipo *int*) e *salario* (tipo *double*), com seus respectivos métodos *getters* e *setters* e também um construtor que invoca o construtor da superclasse.

O salário de um funcionário jamais poderá ser negativo. Todo funcionário recebe seu salário em duas parcelas, sendo 60% na primeira parcela e 40% na segunda parcela. Assim, escreva os métodos:

- a. *getSalarioPrimeiraParcela()* que retorna o valor da primeira parcela do salário (60%);
- b. *getSalarioSegundaParcela()* que retorna o valor da segunda parcela do salário (40%).

3. Crie uma subclasse de *Funcionario*, chamada *Professor*. Todo professor recebe seu salário em uma única parcela. Assim, devem-se sobrescrever os métodos *getSalarioPrimeiraParcela()* e *getSalarioSegundaParcela()*. O método *getSalarioPrimeiraParcela()* da classe *Professor* deve retornar o valor integral do salário do professor e o método *getSalarioSegundaParcela()* do professor deve retornar o valor zero. Crie também um construtor que invoca o construtor da superclasse.

Em seu método *main*, crie objetos das três classes da hierarquia e imprima a qual classe cada objeto pertence, obtendo esta informação através do método *getClass()* da classe *Object*.

4. Crie uma classe *Equipamento* com o atributo *ligado* (tipo *boolean*) e com os métodos *liga* e *desliga*. O método *liga*() torna o atributo *ligado* *true* e o método *desliga*() torna o atributo *ligado* *false*.

Crie também uma classe *EquipamentoSonoro* que herda as características de *Equipamento* e que possui os atributos *volume* (tipo *short*) que varia de 0 a 10 e *stereo* (tipo *boolean*). A classe ainda deve possuir métodos *getters* e *setters*, além dos métodos *mono()* e *stereo()*. O método *mono()* torna o atributo *stereo* falso e o método *stereo()* torna o atributo *stereo* verdadeiro. Ao ligar o *EquipamentoSonoro* através do método *liga*, seu volume é automaticamente ajustado para 5.

5. Crie uma classe *Transporte* com atributos *ligado* (tipo *boolean*) e *velocidade* (tipo *int*) e métodos *liga*() e *desliga*(). O método *liga* torna o atributo *ligado* *true* e o método *desliga* torna o atributo *ligado* *false*, além de tornar a *velocidade* zero. Crie também métodos *getters* e *setters* para modificar o atributo *velocidade*, sendo que a velocidade não pode ser negativa.

Crie uma subclasse de *Transporte* chamada *Carro*. *Carro* deve ter o atributo *quilometragem* (tipo *int*) e os métodos *getters* e *setters*. A quilometragem não pode ser negativa, nem ultrapassar o valor 999999. A velocidade do carro não pode ser negativa, nem ultrapassar 200.

6. Crie uma classe *Conta* com atributo *saldo* e métodos *depositar()* e *sacar()* para diminuir e aumentar o atributo *saldo*, respectivamente, a partir de um valor especificado. O atributo *saldo* pode ser negativo.

Crie uma subclasse de *Conta* chamada *Poupanca*. Esta subclasse deve ter o atributo *diaRendimento* do tipo *int* que armazena o dia do mês em que ocorre o aniversário da poupança. Além disso, ainda deve possuir os métodos *getter* e *setter*. O atributo *saldo* da *Poupanca* não pode ser negativo.

7. Crie classes *Humano*, *Inseto* e *SuperHeroi*. A classe *Humano* possui atributos *Nome*, *Sexo*, *Idade*, *Lingua* e *Etnia*, todos especificados como *protected*. A classe *Inseto* possui atributos *Nome*, *Veneno*, *Alado*, *Ferrao* (os três últimos são *boolean*), todos também especificados como *protected*. A classe *SuperHeroi* herda das duas classes anteriores, e adiciona os atributos *Codig*, *Trauma* e *Poderes*. Os atributos não possuem *getters* e *setters*. Crie a hierarquia de herança um construtor e um método *print* (utilize o *toString*) na subclasse.