

# Lista de Exercícios Herança

## Exercicio 01 - Sistema Bancário

Crie uma superclasse chamada ContaBancaria com os seguintes atributos:

- String nomeCliente
- Integer numeroConta
- double saldo

E os seguintes métodos.

- sacar
- depositar

Estes métodos devem atualizar o atributo saldo. O saldo ao sacar não deve ficar negativo para isso realize a validação e imprima uma mensagem via System.out ou JOptionPane que não foi possível sacar pois o saldo é insuficiente.

Crie também duas subclasses chamadas ContaCorrente e ContaInvestimento, com as seguintes características:

ContaCorrente

- Novo atributo chamado double limite.
- sobrescreva o método sacar com a nova lógica que suporta o atributo limite em que o cliente pode sacar o valor do saldo + o limite.

ContaInvestimento

- Novo atributo chamado int diaRendimento;
- metodo calcularNovoSaldo que recebe o percentual de rendimento mensal da conta de investimento e verifica se o dia de hoje é o diaRendimento e se sim realiza o calculo do novo valor do saldo considerando o percentual de rendimento da conta. Este metodo deve atualizar o valor do atributo saldo e ao final imprima uma mensagem via System.out ou JOptionPane com o novo valor de saldo.

Para buscar o dia do mês utilizem o código abaixo:

```
Date date = new Date();// Pega o dia de hoje
Calendar calendar = Calendar.getInstance();
calendar.setTime(date);
int dia = calendar.get(Calendar.DAY_OF_MONTH);//Pega o dia
do mes a partir da data
```

Crie uma classe principal que contenha a seguinte lógica:

- Crie 2 contas uma de cada tipo.
- Sacar valor das duas contas.
- Depositar o valor nas duas contas.
- Para a ContaInvestimento mostrar o saldo a partir de um investimento.
- Mostrar os dados das contas.

## **Exercicio 02 - Animais**

Um animal contém nome, comprimento, número de patas (padrão é 4), cor, ambiente e velocidade (em m/s).

Um peixe é um animal, tem 0 patas, seu ambiente é o mar (padrão), cor cinzenta (padrão). Além disso, tem como características barbatanas e cauda.

Um mamífero é um animal e seu ambiente padrão é terra.

Um urso é um mamífero, cor castanho e seu alimento preferido é o mel.

Crie as classes Animal, Peixe e Mamífero.

Para a classe Animal, coloque os atributos:

- Nome, comprimento, patas, cor, ambiente, velocidade

Para a classe Peixe, codifique o atributo características.

Para a classe Mamífero, coloque o atributo alimento.

Por último, crie uma classe teste de forma a ter um jardim zoológico com os seguintes animais: camelo, tubarão, urso-do-canadá.

Exemplo de Execução:

Zoo:

-----  
Animal: Camelo  
Comprimento: 150 cm  
Patas: 4  
Cor: Amarelo  
Ambiente: Terra  
Velocidade: 2.0 m/s  
-----

Animal: Tubarão  
Comprimento: 300 cm  
Patas: 0  
Cor: Cinzento  
Ambiente: Mar  
Velocidade: 1.5 m/s  
Característica: Barbatanas e cauda.  
-----

Animal: Urso-do-canadá  
Comprimento: 180 cm  
Patas: 4  
Cor: Vermelho  
Ambiente: Terra  
Velocidade: 0.5 m/s  
Alimento: Mel  
-----

### Exercício 03 - Ingressos

Crie uma classe chamada Ingresso que possui um valor em reais e um método imprimeValor().

- a. crie uma classe VIP, que herda Ingresso e possui um valor adicional. sobrescreva o método imprimeValor() para que retorne o valor do ingresso VIP (com o adicional incluído).
- b. crie uma classe Normal, que herda Ingresso e possui um método que imprime: "Ingresso Normal".

c. crie uma classe CamaroteInferior (que possui a localização do ingresso e método para imprimir esta localização) e uma classe CamaroteSuperior, que é mais cara (possui valor adicional). Esta última possui um método para retornar o valor do ingresso.

Ambas as classes herdam a classe VIP.

### Exercício 4 - Corretora de Imóveis:

Crie a classe Imovel, que possui um endereço e um preço.

- a. crie uma classe Novo, que herda Imovel e possui um adicional no preço. Crie métodos de acesso e impressão deste valor adicional.

**b.** crie uma classe Velho, que herda Imovel e possui um desconto no preço. Crie métodos de acesso e impressão para este desconto.