

## **Lista exercícios - Orientação a objetos**

### **Questão 1**

Identifique as classes, atributos e métodos necessários para modelar e implementar:

- a) Uma conta corrente que possui um número, um saldo, um limite, um status que informa se ela é especial ou não e um conjunto de movimentações;
- b) Uma movimentação que possui uma descrição, um valor e uma informação se ela é uma movimentação de crédito ou débito;
- c) Um banco que armazena um conjunto de contas e fornece métodos que permitam que sejam feitas criações e exclusões de contas, saques (uma conta só pode fazer saques desde que o valor não exceda o saldo + limite), depósitos, emissão de saldo, extrato e transferência entre contas.

### **Questão 2**

Identifique as classes, atributos e métodos necessários para modelar e implementar:

- a) Um país que possui um nome, nome da capital, sua dimensão em Km<sup>2</sup> e uma lista de países com os quais faz fronteira;
- b) Um construtor que inicialize o nome, nome da capital e dimensão do país;
- c) Métodos de acesso (get/set) para as propriedades indicadas no item (a)
- d) Um método que define quais outros países fazem fronteira (Obs: um país não pode fazer fronteira com ele mesmo);
- e) Um método que retorne a lista de países que fazem fronteira;
- f) Um método que receba um outro país como parâmetro e retorne uma lista de vizinhos comuns aos dois países.

### **Questão 3**

De forma incremental, traduza o seguinte conjunto de classes em um programa Java. Não utilize System.in ou System.out dentro das classes criadas. Utilize System.out apenas no método main.

a) Classe: Porta

Atributos: *aberta*, *cor*, *dimensaoX*, *dimensaoY*, *dimensaoZ*

Métodos: *void abre( )*, *void fecha( )*, *void pinta(String s)*, *boolean estaAberta( )*

Para testar, crie uma porta, abra e fecha a mesma, pinte-a de diversas cores, altere suas dimensões e use o método *estaAberta* para verificar se ela está aberta.

b) Classe: Casa

Atributos: *cor*, *porta1*, *porta2*, *porta3*

Método: *void pinta(String s)*, *int quantasPortasEstaoAbertas( )*, *int totalDePortas( )*

Para testar, crie uma casa e pinte-a. Crie três portas e coloque-as na casa; abra e feche as mesmas como desejar. Utilize o método *quantasPortasEstaoAbertas* para imprimir o número de portas abertas.

c) Classe: Edificio

Atributos: *cor*, *totalDePortas*, *totalDeAndares*, *portas[ ]*

Métodos: *void pinta(String s)*, *int quantasPortasEstaoAbertas( )*, *void adicionaPorta(Porta p)*, *int totalDePortas( )*, *void adicionarAndar( )*, *int totalDeAndares( )*

Para testar, crie um edifício, pinte-o. Crie seis portas e coloque-as no edifício através do método *adicionaPorta*, abra e feche-as como desejar. Utilize o método *quantasPortasEstaoAbertas* para imprimir o número de portas abertas e o método *totalDePortas* para imprimir o total de portas em seu edifício. Crie alguns andares utilizando o método *adicionarAndar* e retorne o número total de andares utilizando o método *totalDeAndares*

d) As classes *Casa* e *Edificio* ficaram muito parecidas. Crie a classe *Imovel* e coloque nela tudo o que *Casa* e *Edificio* tem em comum. Faça *Imovel* superclasse de *Casa* e *Edificio*. Note que alguns métodos em comum não poderão ser implementados por *Imovel*. Logo, esses devem ser declarados como métodos abstratos.

#### Questão 4

Crie uma classe abstracta de polígonos. Todo polígono deve saber calcular sua área. Crie três tipos de especializações: um círculo, um retângulo e um triângulo. Sobrescreva nas especializações o método de cálculo de área declarado na classe pai. Crie uma classe de polígono complexo que pode ser constituída por um ou mais polígonos. Forneça um método que calcule a área de polígonos complexos. Utilize o princípio de polimorfismo sempre que possível.