

COTIL- UNICAMP
Linguagem de Programação Multiplataforma
Profa. Tania Basso

Lista de Exercícios 1

- 1) Crie uma classe chamada **Funcionário** que inclua 3 atributos: **nome**, **sobrenome**, **salário mensal**. Sua classe deve ter um construtor que inicialize esses 3 atributos com valores vazios e zeros. Também deve ter a sobrecarga do construtor, que recebe esses 3 atributos como argumento. Forneça métodos **get** e **set** para todos os atributos. Se o salário mensal não for positivo, configure-o como zero. Escreva um programa que demonstre as capacidades da classe **Funcionário**. Crie dois objetos do tipo **Funcionário** e exiba o salário **ANUAL** de cada objeto. Dê a cada funcionário um aumento de 10% e exiba novamente o salário anual de cada um deles.
- 2) Crie uma classe chamada **Item** que uma loja de suprimentos de informática possa utilizar para representar uma fatura de um item vendido na loja. Um **Item** faturado deve ter quatro atributos: **código**, **descrição**, **quantidade comprada do item**, **preço do item**. Sua classe deve ter um construtor que inicializa os quatro atributos. Forneça métodos **get** e **set** para cada um dos atributos. Forneça também um método chamado **getTotal** que calcula e retorna o total a pagar sobre o item faturado (ou seja, quantidade * preço). Se a quantidade não for positiva, deve ser configurada como zero. Idem para preço. Escreva um programa que teste a classe **Item**.
- 3) Crie uma classe chamada Pessoa com as seguintes características:
 - Atributos: **idade**, **dia**, **mês** e **ano de nascimento**, **nome da pessoa**.
 - Métodos: **calculaIdade()**, que recebe a data atual em dias, mês e anos e calcula e armazena no atributo idade a idade atual da pessoa; **informaIdade()**, que retorna o valor da idade; **informaNome()**, que retorna o nome da pessoa; um construtor que deve receber dia, mês e ano de nascimento como parâmetros e preencher nos atributos correspondentes do objeto; outro construtor (sobrecarga), que deve receber todos os atributos como parâmetros (exceto a idade, que deve ser calculada). A classe deve também implementar métodos para **validar** dia (1 a 31), mês (1 a 12) e ano (1000 em diante).Fazer uma classe principal que instancie ao menos dois objetos, inicialize e mostre quais seriam as idades de Einstein e Newton caso estivessem vivos (Albert Einstein nasceu em 14/3/1879 e Isaac Newton nasceu em 4/1/1643).
- 4) Escreva uma classe em Java chamada **Estoque**. Ela deve possuir:
 - a. os atributos **nome** (String), **qtdAtual** (int) e **qtdMinima** (int).
 - b. um construtor sem parâmetros e um outro contendo os parâmetros nome, qtdAtual e qtdMinima.
 - c. Os métodos **get** e **set** para cada atributo, assim como um método para **repor** a quantidade em estoque e outro para **dar baixa** na quantidade em estoque. O método **repor** aumenta a **qtdAtual** de acordo com um parâmetro **qtd**. O método **darBaixa** diminui **qtdAtual** de acordo com um

parâmetro **qtd**. Os atributos **qtdAtual** e **qtdMinima** jamais poderão ser negativos. A classe deve implementar um método **mostra()**, que retorna uma String contendo o nome do produto, sua quantidade mínima e sua quantidade atual. Também deve implementar um método **precisaRepor**, que retorna **true** caso a quantidade atual esteja menor ou igual à quantidade mínima e **false**, caso contrário.

Escreva uma classe cujo método **main** instancia os 3 objetos a seguir:

estoque1 nome: Impressora qtdAtual: 13 qtdMinima: 6	estoque2 nome: Monitor qtdAtual: 11 qtdMinima: 13	estoque3 nome: Mouse qtdAtual: 6 qtdMinima: 2
--	--	--

Depois disso, execute as seguintes operações na seguinte ordem:

- Dar baixa em 5 unidades de estoque1.
- Fazer a reposição de 7 unidades de estoque2.
- Dar baixa em 4 unidades de estoque3.
- Exibir a saída do método **precisaRepor** dos 3 objetos.
- Exibir a saída do método **mostra** para apresentar as informações sobre os 3 objetos.