

Classes e Objetos

Programação Java

Prof. Emerson Lima

Questão 1. (*Classe Pessoa*) Faça o que se pede:

- a. Construa uma classe Pessoa com os atributos nome, email e data de aniversário.
- b. Para cada um desses atributos implemente um método acessor (*getter*) e um método definidor (*setter*).
- c. Implemente um atributo somente leitura *idade* que retorna a idade da pessoa calculando-a a partir da informação da data de aniversário. Veja que não é necessário variável para armazenar a informação da idade, e é preciso apenas um método *getIdade()*.
- d. Sobrecarregue o construtor padrão com um que recebe as informações de nome, email e idade.
- e. Sobreescreva o método *toString()* da classe Pessoa para retornar algo como "nome (email)".
- f. Sobreescreva o método *equals()* da classe Pessoa para comparar seus objetos com outros objetos do universo java. Use a seguinte regra:
 - i. Se o objeto que estou comparando não é um objeto do tipo Pessoa, retorno falso.
 - ii. Se o objeto que estou comparando também é uma instância de Pessoa, então seremos iguais se nossos CPFs forem iguais.
- g. Escreva um programa java para testar sua classe.

Questão 2. (*Inscrição na turma*) Projete classes e objetos para uma aplicação Java que permita o usuário realizar os seguintes casos de uso:

- i. Iniciar a inscrição de alunos em uma turma
- ii. Inscrever alunos em uma turma
- iii. Fechar a inscrição da turma, caso haja mais inscritos que vagas, priorizar os inscritos segundo um critério a ser definido pela diretoria.
- iv. Gerar uma listagem dos alunos selecionados para a turma.

Atenção: Não quero o programa, apenas as classes e algum código que mostre como objetos dessas classes podem ser usados para a construção do programa.

Questão 3. (*Carrinho de compras*) Implemente as classes necessárias para que o código a seguir seja executável.

```
1 ListaProdutos lp = new ListaProdutos();
2 lp.add(new Produto("123456789", "Sabão", 2.78));
3
4 CarrinhoCompras cc = new CarrinhoCompras();
5
6 Produto p = lp.buscarPeloCodigoBarras("123456789"),
7 cc.add(new ItemCarrinho(p, 3));
8
9 Pedido pedido = cc.fecharPedido();
10 pedido.imprimir();
```

- Faça modificações no código para permitir a inclusão da forma de pagamento no pedido.
- Faça modificações no código para que seja possível dar baixa em um estoque.
- Faça modificações no código para que promoções do tipo "leve X unidades pague Y unidades" sejam aplicadas em alguns produtos.
- Faça modificações no código para que promoções do tipo "leve X unidades e receba desconto de X% sobre o valor unitário" sejam aplicadas em alguns produtos.
- Organize esses descontos em uma hierarquia de classes de modo que novos tipos de descontos possam ser adicionados ao sistema sem grandes impactos nas demais classes.

Atenção: Se precisar usar listas use a classe `java.util.ArrayList<T>`