

Exercícios sobre Padrões de Projeto - ProjOO

Forma de realização: **INDIVIDUAL** ou em **DUPLA**

Entregar, via Classroom, os JPGs ou GIFs dos Diagramas de Classe e os seus respectivos códigos-fonte e de teste (que mostram um exemplo da execução da aplicação) até o final da aula de hoje.

Exercício 1:

Na cadeia de restaurantes fast-food *PatternBurgers*, há um padrão para montagem de lanches de crianças. O sanduíche (hambúrguer ou cheeseburger), a batata (pequena, média ou grande) e o brinquedo (carrinho ou bonequinha) são colocados dentro de uma caixa e o refrigerante (coca ou guaraná) é entregue fora da caixa. A classe abaixo é dada para representar o pedido de um consumidor:

```
import java.util.*;

public class Pedido {
    private Set<String> dentroDaCaixa = new HashSet<String>();
    private Set<String> foraDaCaixa = new HashSet<String>();

    public void adicionarDentroDaCaixa(String item) {
        dentroDaCaixa.add(item);
    }

    public void adicionarForaDaCaixa(String item) {
        foraDaCaixa.add(item);
    }

    public String toString() {
        StringBuffer buffer = new StringBuffer();
        buffer.append("Seu pedido:\n");
        buffer.append("Dentro da caixa:\n");
        for (String item : dentroDaCaixa) buffer.append("\t" + item + "\n");
        buffer.append("Fora da caixa:\n");
        for (String item : foraDaCaixa) buffer.append("\t" + item + "\n");
        buffer.append("\nTenha um bom dia!\n\n");
        return buffer.toString();
    }
}
```

Neste caso, o padrão *Builder* pode ser usado para separar as tarefas do atendente e do funcionário que monta o pedido. Somente este último sabe como montar os pedidos segundo os padrões da empresa, mas é o atendente quem lhe informa quais itens o consumidor pediu.

Implemente a simulação do restaurante fast-food descrita acima utilizando o padrão Builder e escreva uma classe cliente que pede um lanche ao atendente, recebe-o do outro funcionário e imprime o pedido.