

1. Durante a execução do Tutorial 2 você anotou as ações necessárias para formar a figura indicada. Agora você irá programar essas ações, ou seja, vai transformar aquela sequência de mensagens passadas diretamente aos objetos por meio do BlueJ em um programa. Por onde começar? Para realizar isso, o que é preciso?
2. Em uma classe denominada Pessoa, deseja-se manter o nome, sexo, data de nascimento e estado civil de uma determinada pessoa. O nome, sexo e data de nascimento devem ser atribuídos no construtor da classe, bem como o estado civil deve ser definido como "solteiro", por padrão. O sexo deve ser definido como um caracter ('F' ou 'M') e o estado civil deve ser definido como um string. Baseado nisto, descreva o código Java da classe Pessoa. Complete os tipos dos atributos e parâmetros.

Pessoa	
<i>Attributes</i>	
private _____	nome
private _____	sexo
private _____	dataNascimento
private _____	estadoCivil
<i>Operations</i>	
public Pessoa( _____ nome, _____ sexo, _____ dataNascimento )	

3. A partir das representações abaixo dos objetos de uma classe Produto, escreva o código Java necessário para suportar tais objetos, ou seja, o código necessário para criar a classe e instanciar os objetos de modo que fiquem com o estado apresentado.

produto1: Produto
nome = "Caderno"
descrição = "Caderno em espiral tamanho médio"
precoUnitario = 4,50
desconto = 15

produto3: Produto
nome = "Esquadro"
descrição = "Esquadro de acrílico 20 cm"
precoUnitario = 2,35
desconto = 10

produto2: Produto
nome = "Caneta ESF"
descrição = "Caneta esferográfica 5mm"
precoUnitario = 1,20
desconto = 2

4. O código a seguir foi escrito por um programador descuidado que, após tê-lo escrito, perdeu o código original da classe Cidade. Verifique o trecho de código abaixo, e escreva qual seria o código necessário da classe (atributos, métodos e construtores).

```
...
Cidade gaspar = new Cidade();
gaspar.setNome( "Gaspar" );
gaspar.setUf( "SC" );
gaspar.setPopulacaoMasculina( 28141 );

Cidade indaial = new Cidade();
indaial.setNome( "Indaial" );
indaial.setPopulacaoFeminina( 25844 );
indaial.setCalculado( true );

Cidade blumenau = new Cidade( "Blumenau", 154580, 165420 );
...
```

5. Um matemático está necessitando de várias funções relacionadas a um número inteiro positivo. Suponha a definição de uma classe Inteiro Positivo que apresenta o seguinte atributo: um número X. Implemente os seguintes métodos:

- um método setValor, que realiza a consistência necessária para garantir que X seja um inteiro positivo;
- um método para retornar o número X multiplicado por outro objeto de *InteiroPositivo*;
- um método para calcular o fatorial de X;  
Fatorial (X) = X \* (X-1) \* (X-2) \* (X-3) \* ... \* 2 \* 1
- um método para identificar os divisores inteiros de X e a quantidade de divisores. Exemplo: para o número 12, os divisores são 1, 2, 3, 4, 6, 12 e a quantidade de divisores é 6;
- um método para calcular a série de Fibonacci formada por X elementos;  
Fibonacci = 1, 1, 2, 3, 5, 8, 13, ...
- um método para retornar o valor de H, baseado na fórmula:

$$H = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{X}, \text{ também expresso por } \sum_{d=1}^X \frac{1}{d}$$

- g) um método para retornar o valor de I, baseado na fórmula:

$$I = \frac{x}{1} - \frac{x-1}{2} - \frac{x-2}{3} - \dots - \frac{2}{x-1} - \frac{1}{x}$$

- h) um método para retornar o valor de S, baseado na fórmula:

$$S = x^{20} + \frac{x^{19}}{2!} + \frac{x^{18}}{3!} + \frac{x^{17}}{4!} + \dots + \frac{x^{16}}{5!} + \dots + \frac{x^5}{16!} + \frac{x^4}{17!} + \frac{x^3}{18!} + \frac{x^2}{19!} + \frac{x}{20!}$$

- i) um método para retornar o valor de P, baseado na fórmula:

$$P = \frac{2!}{1} + \frac{4!}{2} - \frac{6!}{3} + \frac{8!}{4} - \frac{10!}{5} + \frac{12!}{6} \dots, \text{ considerando X termos.}$$