

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO GRANDE DO NORTE

PRO-REITORIA DE ENSINO

Av. Sen. Salgado Filho, 1559, Tirol, Natal/RN, 59015-000.Fone/FAX +55 (84) 4005-2637. URL: www.ifrn.edu.br

### 3ª LISTA DE EXERCÍCIOS SOBRE POO

<b>Aluno:</b> _____	<b>Área Profissional:</b> Informática
<b>Disciplina:</b> Programação Orientada a Objetos	<b>Questões:</b> 5
<b>Professor:</b> Diego Henrique Oliveira de Souza	<b>Tempo:</b> 45 minutos
<b>Data:</b> <u>17</u> / <u>12</u> / <u>2015</u>	<b>Modalidade:</b> Revisão

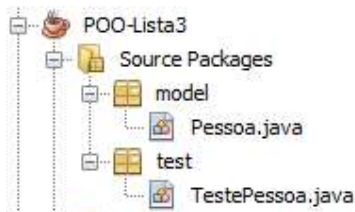
#### QUESTÕES

**1-Construtores:** como ficaria o construtor da classe abaixo para preencher todos os seus atributos?

```
public class Pessoa {
    private String nome;
    private int idade;
}
```

- a) public void Pessoa(String nome, int idade){}
- b) public Pessoa(String nome, int idade){}
- c) private Pessoa(String nome, int idade){}
- d) public Pessoa(nome, idade){}
- e) private Pessoa(String, int){}

**2-Pacotes:** a respeito da estrutura abaixo, marque a alternativa correta:



- a) TestePessoa não verá os atributos public de Pessoa
- b) TestePessoa verá os atributos private de Pessoa
- c) TestePessoa verá os atributos public de Pessoa só se importar a classe Pessoa
- d) TestePessoa verá os atributos private de Pessoa caso importe Pessoa
- e) TestePessoa não tem como ver os atributos de Pessoa

**3-Encapsulamento:** não é um dos modificadores de acesso do java:

- a) default
- b) protected
- c) private
- d) static
- e) public

**4-Herança:** como fazer para uma classe Civic herdar os dados de Sedan, Carro e Veículo?

- a) public class Civic extends Sedan, Carro, Veiculo{}
- b) public class Civic extends Sedan{  
public class Sedan extends Carro{  
public class Carro extends Veiculo{}
- c) public class Veiculo extends Carro{  
public class Carro extends Sedan{  
public class Sedan extends Civic{}
- d) public class Civic extends Sedan & Carro & Veiculo{}
- e) public class Civic extends Sedan extends Carro extends Veiculo{}

**5-Polimorfismo:** uma classe mãe no polimorfismo possui métodos que definem um contrato. Cada classe filha deve implementar estes métodos. São os métodos:

- a) obligated
- b) undefined
- c) contract
- d) concrete
- e) abstract

**6-(1,0 EXTRA):** Qual a saída do código abaixo?

```
public class Teste {
    public static void main(String[] args) {
        int c = 5, r = 4, g = 1;
        for (int j = 0; j < r; j++) {
            g *= c;
            System.out.println(g);
        }
    }
}
```

**RESPOSTA:** \_\_\_\_\_

Java é o poder!  
Boa Sorte!