



++

45697056

Sistemas para Internet

Domain Driven Design

45697056

Sobrecarga e Construtores - Exercícios

45697056

...



Exercício 01

Implemente as seguintes classes:

Funcionario
~ nome : String ~ matricula : long ~ profissao : Profissao ~ salario : double
~ Funcionario() ~ Funcionario(matricula : long) ~ Funcionario(matricula : long, nome : String) ~ Funcionario(matricula : long, nome : String, profissao : Profissao) ~ exibirDadosFuncionario() : void ~ setProfissao(profissao : Profissao) : void

Profissao
~ nome : String

Construa uma classe chamada Veiculo com:

- A. Os atributos: Placa, Marca, Modelo e Ano
- B. Os seguintes métodos:
 - a. Construtor sem argumentos
 - b. Construtor com argumentos
 - c. Métodos get para retornar os valores dos atributos, ex.: getPlaca
 - d. Métodos set para atribuir os valores aos atributos, ex.: setPlaca
 - e. Regra de validação: $\text{Ano} \geq 1900$ e $\text{Ano} \leq 2017$

Escreva uma classe chamada Data que contenha três atributos do tipo int chamados **dia**, **mes** e **ano**

- a) Crie um método construtor que inicialize os dados com zero e outro método construtor que inicialize os dados com os valores recebidos como parâmetros.
- b) Crie um método para retornar a data como string no formato: dd/mm/aaaa
- c) Crie a validação do dia. Ele deve estar entre 1 e 31. Caso contrário inicializar o atributo com o valor 1.
- d) Crie a validação do mês. Ele deve estar entre 1 e 12. Caso contrário inicializar o atributo com 1.
- e) Crie a validação do ano. Ele não deve ser negativo. Se o ano for inválido, defina o valor do atributo ano com o ano corrente.
- f) Crie um método que retorne o nome do mês de acordo com o número que está armazenado no atributo mes.

Crie uma classe chamada **Empregado**, dentro do pacote **br.com.fiap.lista08.exercicio04**.

- a) Crie os atributos **nome**, **funcao** e **salario**
- b) Crie um construtor que receba como parâmetros o nome, função e salário do empregado e depois inicialize os respectivos atributos.
- c) Crie um método chamado **aumentarSalario**, que receberá como parâmetro o valor do aumento. Este método deverá atualizar o salário do empregado.
- d) Crie uma classe de teste chamada **TesteSalario** no mesmo pacote da classe **Empregado**.
 - a. Crie um objeto da classe Empregado que tenha como nome “João”, salário inicial de R\$5.400,00 e função “Analista de Sistemas”. Utilize o construtor criado no **item b**.
 - b. Aumente o salário deste funcionário em R\$600,00, através da utilização do método criado no **item c**.
 - c. Exiba na tela valor dos atributos **nome**, **funcao** e **salario**.

Crie uma classe chamada **TrianguloRetangulo**, no pacote: **br.com.fiap.lista08.exercicio05**.

- a) Crie os atributos **cateto1** e **cateto2**
- b) Declare o construtor padrão para a classe **TrianguloRetangulo**.
- c) Crie um construtor que receberá como parâmetros os catetos(cateto1 e cateto2) e atribuirá esses valores aos respectivos atributos.
- d) Crie um método chamado **calcularHipotenusa**, sem parâmetros de entrada para realizar o cálculo da hipotenusa e retornar o seu tamanho (*ver dica no próximo slide*).
- e) Crie uma classe de teste chamada **TesteTriangulo**.
 - Crie 2 objetos: o primeiro utilizando o construtor padrão, com catetos de tamanho 5 e 10 e o segundo utilizando o construtor de classe criado no item c, com catetos de tamanho 4 e 5.
 - Exiba na tela os valores das hipotenusas.

Exercício 05

Nota: Para efetuar o cálculo da hipotenusa, utilize a fórmula:

$$\text{hipotenusa}^2 = \text{cateto1}^2 + \text{cateto2}^2$$

Para realizar a potenciação, utilize o método `pow` da classe **java.lang.Math**.

Este método recebe dois parâmetros do tipo `double` e retorna o valor obtido da operação.

Por exemplo, para x^y : `double resultado = Math.pow(x, y);`

Para realizar a raiz quadrada, utilize o método `sqrt` da classe **java.lang.Math**.

Este método retorna a raiz quadrada do número informado como parâmetro, o parâmetro de entrada e o retorno são do tipo `double`.

Por exemplo, para raiz quadrada de 16: `double resultado = Math.sqrt(16.0);`

Exercício 06



Crie uma classe chamada **Hora**, no pacote: **br.com.fiap.lista08.exercicio06**

- a) Crie os atributos **hora**, **minuto** e **segundo**
- b) Crie um construtor que receba 3 parâmetros: a hora, o minuto e o segundo e em seguida atribua esses valores aos respectivos atributos da classe.
- c) Crie um método chamado **obterHora**, para retornar a hora, no formato: **hh:mm:ss**
- d) Crie uma classe de teste chamada **TesteHora**, esta classe deve instanciar um objeto da classe Hora e definir seu estado de modo que ao ser invocado o método **obterHora** seja exibido no console a hora 12:40:15

Crie uma classe chamada **Lampada**, no pacote **br.com.fiap.lista08.exercicio07**

1. Crie os atributos **cor** do tipo **String** e **ligada** do tipo **boolean**
2. Crie um construtor sem parâmetro.
3. Crie um construtor que receba 2 parâmetros: a cor e o estado da lâmpada (true ou false) e atribua aos respectivos atributos.
4. Crie um construtor que receba apenas um parâmetro: a cor da lâmpada. Neste caso, atribua a cor ao respectivo atributo e por padrão, atribua ao atributo ligada o valor true.
5. Crie uma classe de teste chamada **TesteLampada** e instancie 3 lâmpadas.
 - a. A primeira deverá ser instanciada através do construtor sem parâmetro
 - b. A segunda, através do construtor com um parâmetro
 - c. A terceira, através do construtor com dois parâmetros
6. Imprima as cores e o estados das lâmpadas instanciadas

Exercício 08

Crie uma classe chamada Ponto, no pacote br.com.fiap.tds.ltp. coordenada, que armazene as coordenadas (x,y) de um ponto num plano. A coordenada default(padrão) para qualquer ponto deve ser (0.0 , 0.0).

- Além do construtor sem parâmetro, esta classe deve conter um construtor que receba como parâmetros os valores de x e y.
- Crie um método para calcular a distância de 2 pontos (x1,y1) e (x2,y2). Este método receberá como parâmetro um objeto da classe Ponto e retornará a distância entre os pontos.
- Crie uma classe de Teste e calcule a distância de 2 pontos e exiba o resultado na tela.
- Use o construtor sem parâmetro para instanciar o primeiro ponto e o construtor com parâmetros para instanciar o segundo ponto.

Nota: Para calcular a distância de 2 pontos utilize a fórmula:

dx = x2 - x1; **dy** = y2 - y1; **distancia** = Math.sqrt(dx*dx + dy*dy);

Crie uma classe chamada **Cor** no pacote **br.com.fiap.lista08.exercicio09**

1. Crie os atributos do tipo inteiro **red**, **green** e **blue**
2. Forneça os construtores apropriados para possibilitar as seguintes instâncias:
 - a. `Cor cor1 = new Cor(10, 100, 20); //red,green,blue`
 - b. `Cor cor2 = new Cor(125, 30); //red,green`
 - c. `Cor cor3 = new Cor(50); //red`
 - d. `Cor cor4 = new Cor();`
3. Crie um método na classe `Cor` chamada `obterCor` que retornará uma `String` no formato:
"`(<red>,<green>,<blue>)`"
4. Crie uma classe de Teste para realizar as instâncias definidas nos itens I, II, III e IV e exibir as cores.

Para mais informações sobre o sistema de cores RGB: <http://pt.wikipedia.org/wiki/RGB>

Exercício 10

Responda:

- 1) Defina sobrecarga de métodos e em seguida implemente uma classe onde é expresso um exemplo.
- 2) O que é um construtor?
- 3) O que é um construtor padrão? Quais as suas características?



Copyright © 2017 Prof. Douglas Cabral <douglas.cabral@fiap.com.br> <https://www.linkedin.com/in/douglascabral/>

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).