Universidade Federal de Uberlândia
Faculdade de Computação - FACOM
FACOM32305 – Programação Orientada a Objetos
Turma 2024/1
Prof. Thiago Pirola Ribeiro

Exercícios sobre Encapsulamento e Construtores

1. Acesse a página disponível em https://www.devmedia.com.br/diferencas-entre-string-stringbuilder-e-stringbuffer-em-java/29865 e leia sobre a diferença entre String, StringBuilder e StringBuffer. Leia com atenção os programas apresentados e os execute. Note a utilização de métodos estáticos e de funções dentro da classe principal. Vocês utilizarão este tipo de método no projeto e nas listas. Não é necessário entregar algo referente a este exercício.

2. Escreva uma classe para representar o nome completo de uma pessoa, composto por três Strings: nome próprio, nome do meio e nome da família. Escreva os métodos descritos a seguir, e implemente a função principal que testa estes métodos.

   a) Construtor com 1 parâmetro, que recebe uma string com o nome completo da pessoa, e separa a String nos três atributos (considere os espaços entre as palavras para a divisão, e assuma que as únicas entradas serão strings com duas ou três palavras).

   b) Construtor com 3 parâmetros, que recebe três strings, cada uma com uma parte do nome.

   c) Construtor sem parâmetros que interage com o usuário e faz a leitura do nome (pode ser separadamente, uma leitura para cada parte do nome).

   d) Um método para mostrar na tela o nome completo, a rubrica e a assinatura da pessoa.

   e) Um método para retornar a rubrica, que consistirá das iniciais do nome completo em caracteres minúsculos.

   f) Um método para retornar a assinatura, que consistirá das iniciais do nome próprio e do meio (com pontos), e o nome de família completo.

   g) Exemplo: Se o nome da pessoa for Julia Cristina Silva, o método rubricar retornará uma string “jcs”, e o método assinar retornará uma string “J. C. Silva”.

   h) Implemente a função principal que cria e manipula pelo menos 3 objetos.

3. Escreva uma classe com os atributos cpf, nome, telefone, usuário e senha que implemente construtores de acordo com:

   a) Recebe três Strings, cpf, nome e telefone. Neste caso, o usuário deve ser determinado como o nome, e a senha deve ser determinada como o cpf.

   b) Recebe quatro Strings, cpf, nome, telefone e e-mail, e define o usuário como sendo o e-mail. É necessário verificar se a String representando o e-mail contém um símbolo @ a partir da segunda posição e .com em alguma posição após o símbolo @. Caso não encontre, deve utilizar o nome como usuário. A senha deve ser o cpf.

   c) Recebe cinco Strings, cpf, nome, telefone, e-mail e senha, e coloca o usuário como o e-mail desde que as condições de @ e .com descritas no item anterior sejam satisfeitas, caso contrário o usuário deve ser o nome, e se a senha contiver menos de 6 caracteres, esta deve ser guardada como o cpf.

d) Implemente também o programa principal e quaisquer métodos necessários para testar a sua classe.

4. Um número racional é um número p/q, onde p e q são inteiros e q é diferente de zero. O número p é chamado de numerador e q de denominador. Implemente uma classe Racional para representar e realizar operações aritméticas em números racionais representados pela classe Racional. A classe deve implementar os seguintes métodos:

   a) Construtor sem parâmetros que deve criar o Racional 0/1.

   b) Construtor com um parâmetro inteiro, que corresponde ao numerador, e o denominador é igual a 1 (lembrando que q é sempre diferente de zero).

   c) Construtor com dois parâmetros inteiros, que correspondem ao numerador e ao denominador. Caso o usuário decida por q nulo, deve ser colocado o valor 1.

   d) exibeRacional: exibe na tela o numerador e o denominador como uma fração (por exemplo, 2/3).

   e) verificaMaiorQue1: escreve na tela se o número racional é maior ou menor que 1.

   f) calcularProduto: calcula o produto de um número racional por outro número racional e retorna este resultado como um número racional (lembrando que o produto de duas frações continua sendo uma fração).

   Implemente a classe principal que cria pelo menos três objetos e testa os métodos disponíveis.