

Entrega: Exercício 1 e Exercício 2 até a letra (c): até o fim da aula
O restante: Até dia 26/04

Exercício 1: Crie uma classe utilitária Estatística. Esta classe é uma classe que irá possuir apenas métodos com o modificador **static**. Não será necessário criar objetos dessa classe.

Crie os seguintes métodos:

a) **media(double[] valores)**: Dado um vetor de valores **double**, calcule (e retorne) a média aritmética.

b) **maximo(double[] valores)**: Calcule o valor máximo entre os valores passados como parametros e retorne-o.

c) **minimo(double[] valores)**: Calcule o valor mínimo entre os valores passados como parametros e retorne-o.

Em Java pode-se haver sobrecarga de métodos. Com isso, um método pode diferenciar apenas pelos seus parâmetros. Faça, então, os métodos: media, minimo e maximo que, dessa vez, utilizam como parâmetro uma lista de double (List<double>) ao invés de um vetor de double.

Exercício 2: Na primeira aula prática, implementamos a classe Atleta. Agora, faça algumas alterações na classe atleta e construa a classe Time.

a) Altere a classe Atleta. Utilize a exceção `IllegalArgumentException` para garantir que o peso e a altura do atleta seja maior que zero e que o nome não seja null.

b) Crie a classe time o seu construtor receberá um vetor com atletas. Utilize a exceção `IllegalArgumentException` para garantir que o vetor não seja null, que não tenha posições null e o número de atletas seja entre 5 e 11.

Na classe Time será necessário também calcular o IMC médio, máximo e mínimo dos atletas. Há várias formas de fazer isso. Para aprendermos a reutilizar o código, iremos fazer de uma forma que, caso seja necessário fazer outra estatística sobre o time, a faríamos mais facilmente:

c) Crie um método **getIMCAtletas()**. Que retorna um vetor com o IMC de todos os atletas do time. Por exemplo, se tivermos cinco atletas e seus imcs forem 23.3, 20.5, 30.2, 30.2, 20. Este método retornará um vetor de double com estes valores.

d) Logo após, crie o método **getMediaIMC()**, **getMaximoIMC()** e **getMinimoIMC()** que retornam a média dos IMCs, o IMC máximo e o mínimo, respectivamente. Estes três métodos utilizarão o método **getIMCAtletas()**, para obter o vetor de IMCs, e um dos métodos da classe Estatística (implementados no exercício anterior) para retornar cada estatística referente ao time.

Dica: como será utilizada uma classe do exercício anterior que está em outro pacote, este

deverá ser importado na classe Time. Para fazer isso, utilize o comando import (no início da classe, antes da declaração. Exemplo:

```
import exercicio1.Estatistica;  
public class Time{  
    ...  
}
```

Caso você escreva na classe time Estatistica.getMedia(imcs). Automaticamente, ele irá fazer o import. Caso não seja feito, pressione: ctrl+shift+O (no Eclipse).

e) Crie o método getAtletaMaiorIMC() que retorna a instância de atleta com o maior IMC.

f) Crie o método **toString** na classe Atleta e nos dados do Time. Apresente, nos dados do atleta, o seu IMC e, nos dados do time, todos os atletas, o imc médio, máximo e mínimo. Além disso, apresente também o nome do atleta com maior imc.

Dica: considere **objAtleta** um objeto do tipo Atleta. Para concatenar os seus dados em um objeto String, podemos simplesmente fazer o seguinte: **“Dados do atleta:”**+objAtleta . Considerando que implementamos o **toString** na classe

g) Finalmente, implemente uma classe de teste que cria um time com 5 atletas para testar a sua implementação.