

Roteiro de Implementação

Você foi contratada(o) para implementar um sistema para uma academia de exercícios físicos. O objetivo é registrar atletas (usuários da academia) e atribuir uma sequência de exercícios (chamada de programa) para cada atleta. Além disso, a academia fornece aos atletas as informações de calorias gastas nos respectivos exercícios, e a sua informação de peso ideal. Siga os passos abaixo para realizar a implementação.



Passo 1: O primeiro passo é permitir que sua academia possa manipular **exercícios**. Cada exercício da sua academia tem um **nome** e uma quantidade inteira de calorias que corresponde à quantidade de **calorias queimadas por hora** daquele exercício. Por exemplo, uma corrida gasta 700 kcal por hora. Diante disso, é possível **consultar** as informações do exercício (nome e kcal/h) bem como descobrir quantas calorias posso queimar em uma **quantidade específica de horas inteiras**. Por exemplo, 3 horas de corrida queimam 2100 kcal.

Exemplo:

Exercicio1 = (Corrida, 700 kcal/h)

Exercicio2 = (Dança, 210 kcal/h)

Passo 2: Agora que podemos criar exercícios, implemente um **programa de exercícios** que é uma sequência de exercícios que deve ser executada. Cada programa possui **no máximo 3 exercícios**. Ao **adicionar exercícios no programa**, sua implementação deve indicar, por meio de um booleano, se o exercício foi adicionado com sucesso. Não é possível adicionar exercícios se:

1. O programa estiver cheio (i.e. já possui três exercícios);
2. O programa já possui um exercício com o mesmo nome;

Exemplo: Programa 1 = (Corrida, 700kcal/h), (Dança, 210 kcal/h)

Passo 3: Agora que podemos adicionar exercícios no programa, vamos melhora-lo para calcular a **quantidade total de kcals gastas** em uma quantidade específica de horas realizando o

programa. Essa quantidade total é a **soma** da quantidade de cada exercício para as X horas especificadas.

Exemplo para o total calorias:

Programa 1: 1 hora = 910 kcal

Programa 1: 3 horas= 2730 kcal

Passo 4: Agora vamos implementar os **atletas** da nossa academia. Cada atleta possui um nome, altura (em metros), peso (em kg), um sexo biológico e um programa (que **inicia vazio** para todo atleta criado). **Cada atleta pode adicionar exercícios ao seu programa.** No entanto, o atleta deve informar se conseguiu adicionar o exercício específico no seu programa, ou não (de acordo com as restrições especificadas no Passo 2). Cada atleta deve também saber **classificar seu IMC de acordo com seu sexo biológico.** O cálculo do IMC é:

$$IMC = massa / (altura)^2$$

Para isso, a atleta deve informar uma das Strings abaixo correspondente à sua **condição de IMC**. A classificação simplificada do IMC está abaixo:

Condição	IMC (Sexo Feminino)	IMC (Sexo Masculino)
Abaixo do peso	<= 19.1	<= 20.7
Intermediário	19.1 a 32.3	20.7 a 31.1
Obeso	>= 32.3	>= 31.1

Passo 5: Cada atleta deve saber representar seu programa de exercícios como uma String da seguinte forma:

Atleta: <nome atleta> ; Peso: <peso> kg; Altura: <altura> m

Programa de exercícios:

==> Corrida: 700 kcal/h

==> Dança: 210 kcal/h

Passo 6: Determine se dois atletas são iguais se eles possuírem **apenas o mesmo programa**. Dois programas são iguais se eles possuem a **mesma quantidade de exercícios** e os **respectivos exercícios na mesma ordem e com o mesmo nome**.

Por exemplo:

Programa 1: “Corrida”; “Dança”

Programa 2: “Corrida”; “Dança”

Programa 3: “Dança”; “Corrida”;

(Programa 1 = Programa 2) != Programa 3

Dica: Crie um método que **retorna o nome de um exercício**, do programa, em um índice específico “i”. **OBS:** se o índice passado não estiver associado a um exercício retorne uma string vazia.

Questao EXTRA (1.0 ponto)

Modifique o sua(s) classes para permitir que um programa tenha qualquer número inteiro positivo de exercícios. Assim, sempre deve ser possível adicionar um novo exercício a um programa. Além disso, adicione uma funcionalidade de ordenar os exercícios de um programa do mais intenso ao menos intenso.

Entrega da prova: Instruções disponíveis no [Canvas](#).

Link para o arquivo de testes: <https://goo.gl/mjFlwS>

Importante:

- Utilize a padronização de nomes em Java para variáveis, classes, métodos e pacotes;
- Implemente focando em modularização e reuso de seu código;
- Divida bem as responsabilidades de suas classes.

Boa prova e boa implementação!