

Você foi contratada(o) para implementar um sistema de uma academia. Esse sistema utiliza arquivos, calcula informações de IMC dos usuários e oferece diferentes opções de pagamento de acordo com a quantidade de exercícios realizadas pelos usuários. Para fornecer uma boa interação com o usuário, forneça um bom tratamento de mensagens de erro, utilizando hierarquia e tratamento de Exception.

- a) Crie o encapsulamento para um usuário da academia. Cada usuário possui um nome, peso, altura e uma lista de exercícios que ele ou ela realizam no seu dia-a-dia a academia. Um usuário não pode mais que 5 exercícios. Trate esses casos lançando um erro com a mensagem: "<nome> ja possui 5 exercicios."
- b) Crie a academia que vai conter os seus Usuários, ou integrantes. A academia possui o método `academia.importaAcademia(String nomeArquivo)` para ler os integrantes e seus respectivos dados a partir de um arquivo no formato CSV (valores separados por vírgula). Cada linha desse arquivo possui as informações de um usuário que deve ser cadastrado na academia. As informações são: nome, altura, peso, e uma lista de exercícios realizada pelo usuários.

<nome>, <altura>, <peso>, <exercício_1>, ..., <exercício_n>

Para verificar se seu método funcionou crie um método para buscar um Usuário no cadastro de sua academia por meio do nome do usuário. Uma vez que recuperar o usuário, verifique se seus atributos foram corretamente armazenados. Se um usuário não for encontrado na sua academia, ao realizar a busca, lance um erro com a seguinte mensagem: "<nome> nao esta cadastrado na academia."

Permita também que um usuário possa adicionar ou remover exercícios a sua lista, por meio dos métodos: `adicionaExercicio(String nomeUsuario, String nomeExercicio);` e `removeExercicio(String nomeUsuario, String nomeExercicio)`. Se o usuário não possui o exercício a ser removido, lance um erro com a mensagem: "<nome> nao possui o exercicio <exercício>."

- c) Cada usuário tem um tipo específico de acordo com a quantidade de exercícios realizada. Se o usuário realiza menos que três exercícios, ele é do tipo Normal, enquanto que os usuários que realizam mais que três ou mais exercícios são do tipo Atleta.

O tipo diferencia no preço pago na mensalidade da academia, bem como no cálculo do IMC do usuário. Os detalhes seguem abaixo:

Usuario Normal:

- $imc = peso / (altura)^2$

- Mensalidade: É pago a mensalidade Padrão (R\$100) com um adicional de R\$70,00 se o usuário fizer o exercício “Pilates”.

Usuario Atleta:

- $imc = 0.9 * (peso / (altura)^2)$
- $mensalidade = R\$100,00 + (0.05 * qtdExercicios * 100)$

Cada tipo possui também um método que deve retornar o seu respectivo tipo: “Usuario” ou “Atleta”.

- d) Cada usuário pode imprimir sua ficha contendo um resumo de suas informações. O formato para imprimir segue abaixo:

Ficha <tipo>: <nome> Peso: <peso>kg Altura: <altura>m. Exercicios praticados:
<exercício_1> <exercício_2> ... <exercício_n>”

Ex: Ficha Usuario: Ronda Peso: 61,00kg Altura: 1,70m. Exercicios praticados:
Malhao Corrida”

- e) Agora que as informações estão no sistema, escreva o método `salvaDados(arquivoDestino)` que salva as fichas de todos os usuários do sistema em um arquivo no formato txt.

Extra (1,0): Mais testes, Javadoc, Criação e uso correto de Exception, ordenar os usuários pelo seu IMC.

Link para a pasta de reposição: <https://goo.gl/674DM5>

Instruções para a entrega:

A entrega será para o e-mail: juiz.lp2@gmail.com

Nomeie seu projeto de acordo com o formato: **Nome_Sobrenome_Matricula**

Envie um zip do seu projeto. Cuidado para empacotar a pasta correta. Confira.

Formato: NOME_SOBRENOME_MATRICULA_TURMA_3_REP_3.zip

No assunto do e-mail coloque: **[Reposicao 3 - 2015.1] Nome Completo**