

## AValiação CONTINUADA 1 – TRABALHO INDIVIDUAL

### Orientações:

- a) Crie um repositório no github com o nome que quiser e vá subindo tudo o que for fazendo nele. Na entrega, apenas cole aqui a URL desse repositório. Você pode continuar fazendo commit no repositório, mesmo após o prazo de entrega. Na avaliação, consideraremos o último commit anterior ao prazo de entrega.
- b) O repositório mencionado acima deverá ser privado. Adicione como membros desse repositório os professores Célia (celiatani) e Yoshi (jyoshiriro-bandtec)
- c) Atenção: Não serão aceitos trabalhos enviados após o prazo de entrega.
- d) Atenção: Resoluções que apresentem evidências de cola terão nota zero.
- e) Atenção: Mesmo tendo compartilhado o link do git com os professores, não deixe de entregar o link no Moodle para que sua entrega fique registrada!

**Questão única:** Crie uma REST API com Spring Boot a qual:

- a) Tenha pelo menos 1 classe abstrata ou 1 interface que obrigue classes diferentes a implementarem um ou mais métodos. Deve haver pelo menos 2 classes que implementam esse(s) método(s). É importante que o método a ser implementado tenha algum cálculo, que é retornado para quem chama, e esse cálculo deve ser diferente em cada classe. Coloque um comentário no código, justificando porque usou classe abstrata ou interface.
- b) Contenha um EndPoint de consulta com método **GET** e um **path param**.
- c) Esse EndPoint devolve uma lista de JSONs criada a partir de uma lista na classe Controller. usada para implementar o conceito de agregação. Essa lista deve ser declarada como sendo do tipo da classe abstrata ou da interface.
- d) O path param vai definir qual implementação da classe abstrata ou lista será usada nos itens da lista. A lista deve ser preenchida no momento da primeira invocação do método desse Endpoint. Caso ela já tenha sido iniciada, o EndPoint apenas devolve a lista que já existe.
- e) Deve haver um EndPoint de exclusão com método **DELETE** que, recebe um **path param** indicando a ordem do item na lista a ser excluído. Ex: 1 para o primeiro item. 2 para o segundo item e assim por diante. Caso seja informado uma ordem que não exista, o Endpoint naturalmente vai retornar um erro.
- f) Se preferir, pode implementar mais EndPoints de consulta, além do especificado acima.