FuturoDEV

Exercícios da Semana

# COMO REALIZAR OS EXERCÍCIOS

Recomenda-se criar uma pasta no computador para armazenar as respostas dos exercícios semanais. No primeiro módulo do curso, os alunos irão receber os enunciados através de um documento texto, após (segundo módulo), os exercícios serão disponibilizados semanalmente em um quadro do *Trello*, onde os alunos irão movimentar os cartões de exercícios/tarefas, seguindo a metodologia *Kanban*, e armazenar as respostas em um repositório no *GitHub*.

# [M1S05] Ex 1 - Validar CPF e CNPJ através de @RequestParam

Através dos conhecimentos adquiridos, disponibilize dois recursos REST para validação de CPF e CNPJ.

# [M1S05] Ex 2 - Calcular média de notas através de @RequestParam

Disponibilize um recurso REST para cálculo de média de notas, utilize 3 notas do tipo Double e retorne a média (Double).

# [M1S05] Ex 3 - Instancie objeto Aluno através de @RequestParam e @RequestBody ( retorne objeto Aluno )

Crie uma classe com nome Aluno, os atributos serão :

* matricula (Long);
* nome (String);

Após disponibilize um recurso REST para instanciar um Aluno através do método POST e retorne Status *CREATED* juntamente com objeto Aluno instanciado.

# [M01S05] Ex 04 - Criar aplicação Java no IntelliJ e utilizando REST no Insomnia que funcione o Get, Post, Put, Delete para criar, consultar, alterar e excluir alunos. Assim como a feita em aula ([código do IntelliJ modelo](https://github.com/maretoli/java-spring-rest-insomnia-crud)).

1. Passa a passo do que fazer:
   1. No site [start.spring.io](http://start.spring.io) crie um aplicação Spring com as dependências Spring Web, H2 Database, Spring Data JPA e Lombok. Baixe o zip do projeto e abra em seu IntelliJ. Execute sua aplicação e valide em seu Terminal se o Spring iniciou normalmente, assim como, verificando em qual porta foi executada a aplicação — geralmente na porta 8080 (<http://localhost:8080/>)
   2. Dentro da pasta de sua aplicação crie dos pacotes:
      1. Pacote: modelo
         1. Classe: Aluno.java
            1. Atributos: matricula (Long e private), nome, telefone, mail (string e private), dataCadastro (Instant e private);
            2. Coloque as anotations: @AllArgsConstructor e @Data e faça os imports do lombok;
      2. Pacote: controller
         1. Classe: AlunoController.java
            1. Faça funcionar com o Insomnia o GET, POST, PUT E DELETE;
   3. Crie no Insomnia e faça todos os testes, garantindo que o GET, POST, PUT E DELETE estejam funcionando.