

Exercícios sobre a primeira versão do Sistema de Controle de Corridas (back end)

Parte 2

- 1) Implemente uma nova funcionalidade (defina um “endpoint” correspondente) que permita consultar as seguintes estatísticas: média, mediana e desvio padrão do tempo em minutos que o corredor tem levado para concluir provas de uma determinada distancia que é informada por parâmetro em uma requisição “GET”. O “endPoint” deve retornar à quantidade de provas consideradas juntamente com as três estatísticas. A estrutura do método que trata tal “endpoint” fica como segue:

```
@GetMapping("/estatisticas")
@CrossOrigin(origins = "*")
public EstatísticasDTO estatisticas(@RequestParam final int distancia){
    ...
}
```

Exemplo de requisição:

<http://localhost:8080/CtrlCorridas/estatisticas?distancia=5000>

- 2) Explique por que foi necessário o uso do padrão DTO para a solução da questão 1.
- 3) Implemente uma nova funcionalidade (defina um “endpoint” correspondente) que permita identificar, dentre as provas da distância indicada ocorridas em um determinado ano (informado pelo usuário), em que momento foi obtido o melhor ganho de performance em relação a prova anterior. Para tanto compare os resultados das provas do ano duas a duas na ordem cronológica e identifique o par onde ocorreu a maior redução no tempo (melhoria de performance). Retorne o nome das duas provas. A estrutura do método que trata tal “endpoint” fica como segue:

```
@GetMapping("/aumentoPerformance")
@CrossOrigin(origins = "*")
public PerformanceDTO aumentoPerformance(@RequestParam final int distancia,
                                           @RequestParam final int ano){
    ...
}
```

Exemplo de requisição:

<http://localhost:8080/CtrlCorridas/aumentoPerformance?distancia=5000&ano=2019>