

Lista 27 – *Redirect vs. Forward*

Na aula 60, do dia 02.03, foi corrigido o programa da calculadora (lista 25). Os métodos utilizam várias classes e métodos diferentes para implementar o redirecionamento de chamadas, com as técnicas de *redirect* e *forward*.

Esse programa possui duas classes que implementam os controladores listados abaixo:

- **CalculadoraRestController, com os *end-points*:**
 - /somar-query
 - /somar-path/{numero1}/{numero2}
 - /subtrair-query
 - /subtrair-path/{numero1}/{numero2}
 - /multiplicar-query
 - /multiplicar-path/{numero1}/{numero2}
 - /dividir-query
 - /dividir-path/{n1}/{n2}
 - /calcular-media-query
 - /calcular-media-path/{numero1}/{numero2}
 - /calcular-potencia-query
 - /calcular-potencia-path/{base}/{expoente}
 - /calcular-raiz-query
 - /calcular-raiz-path/{numero}
- **CalculadoraController, com os *end-points*:**
 - /soma-query
 - /soma-path/{numero1}/{numero2}
 - /soma-forward-query
 - /soma-forward-path/{numero1}/{numero2}

1. Na classe `CalculadoraRestController` faça o seguinte:

- a) Modifique o método `somarQuery` de forma que ela redirecione a chamada feita para o *end-point* `/somar-query` para o *end-point* `/somar-path` usando para isso as classes `RedirectView` e `RedirectAttributes`. Use `RedirectAttributes` para criar 2 atributos quaisquer dentro do método `somarQuery`. Use a classe `Model` e a anotação `@ModelAttribute` para ler estes atributos dentro de `somarPath` e imprimi-los no *console*.
- b) Modifique o método `calcularPotenciaQuery` de forma que ela redirecione a chamada feita para o *end-point* `/calcular-potencia-query` para o *end-point* `/calcular-potencia-path` usando para isso as classes `ModelAndView` e `ModelMap`. Use `ModelMap` para criar 2 atributos quaisquer dentro do método `calcularPotenciaQuery`. Use a classe `Model` e a anotação `@ModelAttribute` para ler estes atributos dentro de `calcularPotenciaPath` e imprimi-los no *console*.
- c) Modifique o método `calcularRaizQuery` de forma que ela faça um encaminhamento (*forward*) de uma chamada feita para o *end-point* `/calcular-raiz-query` para o *end-point* `/calcular-raiz-path` usando para isso as classes `ModelAndView` e `HttpServletRequest`. Use `HttpServletRequest` para criar 2 atributos quaisquer dentro do método `calcularRaizQuery`. Use a classe `HttpServletRequest` para ler estes atributos dentro de `calcularRaizPath` e imprimi-los no *console*.

2. Na classe `CalculadoraController` faça o seguinte:

- a) Siga o modelo do método `somaQuery` (*end-point* `/soma-query`) e implemente o método `subtraiQuery` (*end-point* `/subtrai-query`) que receba 2 números via *query params* e redirecione a chamada para o *end-point* `/subtrai-path` (método `subtraiPath`). O método `subtraiPath` irá imprimir os valores dos números no *console* e no final irá abrir a *view* 'resultado' para mostrar o resultado da subtração na página web.
- b) Siga o modelo do método `somaForwardQuery` (*end-point* `/soma-forward-query`) e implemente o método `multiplicaQuery` (*end-point* `/multiplica-query`) que receba 2 números via *query params* e encaminha a chamada via *forward* para o *end-point* `/multiplica-path` (método `multiplicaPath`). O método `multiplicaPath` irá imprimir os valores dos números no *console* e no final irá abrir a *view* 'resultado' para mostrar o resultado da subtração na página web.