



# Circuit Breaker Pattern

## Vamos praticar!

Nosso chapter lead nos pede para implementar o padrão Circuit Breaker na comunicação de dois microsserviços que estão atualmente registrados em Eureka e se comunicam usando Feign.

Dados os seguintes microsserviços, use Spring Cloud Circuit Breaker em conjunto com Resilience4j para implementar o padrão Circuit Breaker na comunicação entre product-service e review-service.

- eureka-server
- review-service
- product-service

API Doc:

Review-service

[GET] /review/find

Request Params

- productId (obrigatório)
- faultPercent (probabilidade de que o API retorne um erro). (OPCIONAL)

O parâmetro faultPercent é usado para acionar um erro, a fim de testar o Circuit Breaker. É um campo opcional, por padrão a probabilidade de erro é 0.

### Exemplos:

Request:

[GET]

<http://localhost:8084/review/find?productId=1&faultPercent=0>

faultPercentual ajustado para 0 indica que temos 0% de chance de que o API lance um erro.

### Response:



```
[
  {
    "id": 1,
    "title": "Bom produto",
    "detail": "Boa relação/preço qualidade",
    "productId": 1,
    "userId": 1
  },
  {
    "id": 2,
    "title": "Eu amei!",
    "detail": "Excelente qualidade",
    "productId": 1,
    "userId": 2
  }
]
```

Request:

[GET]

<http://localhost:8084/review/find?productId=1&faultPercent=0>

faultPercentual de erro em 99 indica que temos 99% de chance de que o API nos dê um erro.

Response:

```
{
  "timestamp": "2022-03-07T02:57:10.264+00:00",
  "status": 500,
  "error": "Internal Server Error",
  "path": "/review/find"
}
```