README.md 2025-06-06

Exercício Avaliativo: Consumo de API com Feign + DTO + Validação

(Versão Simplificada - 1 Endpoint)



Criar uma aplicação Spring Boot que consuma o endpoint de **produto por ID** da API DummyJSON, utilizando **Feign Client**, **DTOs**, e **validação de parâmetros**.

Requisitos

1. Feign Client

• Implemente um FeignClient para consumir:

```
GET https://dummyjson.com/products/{category}
```

 O cliente deve se chamar ProductClient e estar separado em um pacote apropriado (client ou feign).

2. DTOs

Crie um ProductRequestDT0 com os campos de entrada:

```
public class ProductRequestDTO {
    private double price;
    private int stock;
    private String category;
}
```

Crie um ProductResponseDTO com os campos relevantes do JSON retornado, por exemplo:

```
public class ProductResponseDTO {
    private int id;
    private String title;
    private String description;
    private double price;
    private double rating;
    private int stock;
    private String brand;
    private String category;
}
```

README.md 2025-06-06

3. Validação

- No endpoint da sua aplicação, valide:
 - o category é obrigatório.
 - o stock deve ser maior que zero.
 - o price deve ser maior que zero e menor que 2000.
 - o Caso contrário, retorne erro 400 (Bad Request) com uma mensagem adequada.

4. Service Layer

• Crie a classe ProductService com o método:

```
public ProductResponseDTO getProduct(ProductRequestDTO request)
```

- o O método deve chamar o FeignClient e repassar a categoria.
- Após buscar os produtos, filtre pelo preço maximo e pelo estoque.

5. Controller

• Crie um endpoint:

```
POST /products
```

- Retorna uma lista de ProductResponseDTO dos produtos correspondentes.
- Valide a entrada usando @Requestbody com validação.
- Exemplo de resposta esperada:

```
"id": 1,
  "title": "iPhone 9",
  "description": "An apple mobile which is nothing like apple",
  "price": 549,
  "rating": 4.69,
  "stock": 94,
  "brand": "Apple",
  "category": "smartphones"
}
```

6. Tratamento de Erros

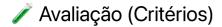
- Se o DTO for inválido, retorne 400 Bad Request.
- Se nenhum produto for encontrado, retorne 404 Not Found com uma mensagem como:

README.md 2025-06-06

Produtos da {category} com estoque maior ou igual a {stock} e valor menor que {price}.

7. Logs

• Crie mecanismo para "printar" o tempo da requisição.



Critério 	Peso
Estrutura do projeto (clean code)	15%
Utilização correta do Feign	25%
Criação e uso de DTO	20%
Aplicação de validações	20%
Tratamento de erros	20%

P Dicas

• Use @FeignClient(name = "productClient", url = "https://dummyjson.com").