

Tutorial

Introdução

Este guia descreve como utilizar a aplicação de gestão de produtos desenvolvida em Java utilizando o framework Spring Boot. A aplicação permite criar, ler, atualizar e deletar produtos.

Pré-requisitos

Para utilizar esta aplicação, você precisará:

- Java 17 ou superior.
- Spring Boot 2.7.3 ou superior.
- Um ambiente de desenvolvimento configurado (IntelliJ IDEA, Eclipse, etc.).
- Banco de dados configurado e conectado à aplicação.

Endpoints Disponíveis

A aplicação possui cinco endpoints principais para gerenciar os produtos, com a seguinte visão geral:

- Criar um Produto (POST /products)
- Listar Todos os Produtos (GET /products)
- Obter um Produto pelo ID (GET /products/{id})
- Atualizar um Produto pelo ID (PUT /products/{id})
- Deletar um Produto pelo ID (DELETE /products/{id})

Criar um Produto

Endpoint: `POST /products`

Descrição: Cria um novo produto na aplicação.

Requisição:

- Método: POST
- URL: /products
- Corpo da Requisição (JSON):

```
{
  "name": "Nome do Produto",
  "description": "Descrição do Produto",
  "price": 100.0
}
```

Resposta:

- Código de Status: 201 CREATED
- Corpo da Resposta (JSON):

```
{
  "id": "UUID do Produto",
  "name": "Nome do Produto",
  "description": "Descrição do Produto",
  "price": 100.0,
  "_links": {
    "self": {
      "href": "http://localhost:8080/products/UUID"
    }
  }
}
```

Listar Todos os Produtos

Endpoint: GET /products

Descrição: Retorna uma lista de todos os produtos.

Requisição:

- Método: GET
- URL: /products

Resposta:

- Código de Status: 200 OK
- Corpo da Resposta (JSON):

```
[
  {
    "id": "UUID do Produto",
    "name": "Nome do Produto",
    "description": "Descrição do Produto",
    "price": 100.0,
    "_links": {
      "self": {
        "href": "http://localhost:8080/products/UUID"
      }
    }
  },
  ...
]
```

Obter um Produto pelo ID

Endpoint: GET /products/{id}

Descrição: Retorna as informações de um produto específico pelo seu ID.

Requisição:

- Método: GET
- URL: /products/{id}
- Parâmetro de Caminho: id (UUID do Produto)

Resposta:

- Código de Status: 200 OK (se o produto for encontrado)
- Código de Status: 404 NOT FOUND (se o produto não for encontrado)

- Corpo da Resposta (JSON):

```
{
  "id": "UUID do Produto",
  "name": "Nome do Produto",
  "description": "Descrição do Produto",
  "price": 100.0,
  "_links": {
    "self": {
      "href": "http://localhost:8080/products/UUID"
    },
    "Lista de Produtos": {
      "href": "http://localhost:8080/products"
    }
  }
}
```

Atualizar um Produto pelo ID

Endpoint: PUT /products/{id}

Descrição: Atualiza as informações de um produto específico pelo seu ID.

Requisição:

- Método: PUT
- URL: /products/{id}
- Parâmetro de Caminho: id (UUID do Produto)
- Corpo da Requisição (JSON):

```
{
  "name": "Nome do Produto Atualizado",
  "description": "Descrição do Produto Atualizada",
  "price": 150.0
}
```

Resposta:

- Código de Status: 200 OK (se o produto for encontrado e atualizado)
- Código de Status: 404 NOT FOUND (se o produto não for encontrado)

- Corpo da Resposta (JSON):

```
{
  "id": "UUID do Produto",
  "name": "Nome do Produto Atualizado",
  "description": "Descrição do Produto Atualizada",
  "price": 150.0,
  "_links": {
    "self": {
      "href": "http://localhost:8080/products/UUID"
    }
  }
}
```

Deletar um Produto pelo ID

Endpoint: `DELETE /products/{id}`

Descrição: Deleta um produto específico pelo seu ID.

Requisição:

- Método: `DELETE`
- URL: `/products/{id}`
- Parâmetro de Caminho: `id` (UUID do Produto)

Resposta:

- Código de Status: `200 OK` (se o produto for encontrado e deletado)
- Código de Status: `404 NOT FOUND` (se o produto não for encontrado)
- Corpo da Resposta (Texto):

```
Produto deletado com sucesso.
```

Utilizando a Aplicação

Para utilizar a aplicação, siga os passos detalhados:

- **Iniciar a Aplicação**

1. **Configuração do Ambiente:**

- Certifique-se de que o Java 17 ou superior está instalado no seu sistema. Você pode verificar isso executando o comando `java -version` no terminal.
- Certifique-se de que seu banco de dados (nesse projeto, está sendo usado o H2, que seria um banco de dados em memória, mas poderia ser, por exemplo, o MySQL, PostgreSQL, etc.) está em execução e configurado corretamente. Verifique as configurações de conexão no arquivo `application.properties` ou `application.yml` do seu projeto Spring Boot.

2. **Compilar e Executar a Aplicação:**

- Abra seu ambiente de desenvolvimento (IDE) onde o projeto está configurado.
- Compile e execute a aplicação. No IntelliJ, você pode fazer isso clicando no botão de "Run" no menu superior.
- A aplicação Spring Boot deve iniciar e estará pronta para receber requisições. Por padrão, ela estará disponível em `http://localhost:8080`.

- **Interagir com os Endpoints**

Para interagir com os endpoints da aplicação, você pode usar ferramentas como **Postman** ou **cURL**. Abaixo estão os passos detalhados para cada operação:

Criar um Produto

Descrição: Este endpoint cria um novo produto na aplicação.

Passos:

1. **Endpoint:** `POST /products`
2. **Método:** `POST`
3. **URL:** `http://localhost:8080/products`

4. Cabeçalhos:

- o Content-Type: application/json

5. Corpo da Requisição (JSON):

```
{  
  "name": "Nome do Produto",  
  "description": "Descrição do Produto",  
  "price": 100.0  
}
```

6. Resposta Esperada:

- Código de Status: 201 CREATED
- Corpo da Resposta (JSON):

```
{  
  "id": "UUID do Produto",  
  "name": "Nome do Produto",  
  "description": "Descrição do Produto",  
  "price": 100.0,  
  "_links": {  
    "self": {  
      "href": "http://localhost:8080/products/UUID"  
    }  
  }  
}
```

Listar Todos os Produtos

Descrição: Este endpoint retorna uma lista de todos os produtos cadastrados na aplicação.

Passos:

1. **Endpoint:** GET /products
2. **Método:** GET
3. **URL:** http://localhost:8080/products
4. **Cabeçalhos:**
 - o Não é necessário cabeçalhos específicos.
5. **Resposta Esperada:**
 - o Código de Status: 200 OK

- Corpo da Resposta (JSON):

```
{
  "id": "UUID do Produto",
  "name": "Nome do Produto",
  "description": "Descrição do Produto",
  "price": 100.0,
  "_links": {
    "self": {
      "href": "http://localhost:8080/products/UUID"
    }
  }
},
...
```

Obter um Produto pelo ID

Descrição: Este endpoint retorna as informações de um produto específico pelo seu ID.

Passos:

1. **Endpoint:** `GET /products/{id}`
2. **Método:** GET
3. **URL:** `http://localhost:8080/products/{id}`
 - Substitua `{id}` pelo UUID do produto que você deseja consultar.
4. **Cabeçalhos:**
 - Não é necessário cabeçalhos específicos.
5. **Resposta Esperada:**
 - Código de Status: 200 OK (se o produto for encontrado)
 - Código de Status: 404 NOT FOUND (se o produto não for encontrado)
 - Corpo da Resposta (JSON):


```
{
  "id": "UUID do Produto",
  "name": "Nome do Produto",
  "description": "Descrição do Produto",
  "price": 100.0,
  "_links": {
    "self": {
      "href": "http://localhost:8080/products/UUID"
    },
    "Lista de Produtos": {
      "href": "http://localhost:8080/products"
    }
  }
}
```

Atualizar um Produto pelo ID

Descrição: Este endpoint atualiza as informações de um produto específico pelo seu ID.

Passos:

1. **Endpoint:** `PUT /products/{id}`
2. **Método:** PUT
3. **URL:** `http://localhost:8080/products/{id}`
 - o Substitua `{id}` pelo UUID do produto que você deseja atualizar.
4. **Cabeçalhos:**
 - o `Content-Type: application/json`
5. **Corpo da Requisição (JSON):**

```
{
  "name": "Nome do Produto Atualizado",
  "description": "Descrição do Produto Atualizada",
  "price": 150.0
}
```

6. Resposta Esperada:

- Código de Status: 200 OK (se o produto for encontrado e atualizado)
- Código de Status: 404 NOT FOUND (se o produto não for encontrado)

- Corpo da Resposta (JSON):

```
{
  "id": "UUID do Produto",
  "name": "Nome do Produto Atualizado",
  "description": "Descrição do Produto Atualizada",
  "price": 150.0,
  "_links": {
    "self": {
      "href": "http://localhost:8080/products/UUID"
    }
  }
}
```

Deletar um Produto pelo ID

Descrição: Este endpoint deleta um produto específico pelo seu ID.

Passos:

1. **Endpoint:** `DELETE /products/{id}`
2. **Método:** DELETE
3. **URL:** `http://localhost:8080/products/{id}`
 - Substitua `{id}` pelo UUID do produto que você deseja deletar.
4. **Cabeçalhos:**
 - Não é necessário cabeçalhos específicos.
5. **Resposta Esperada:**
 - Código de Status: 200 OK (se o produto for encontrado e deletado)
 - Código de Status: 404 NOT FOUND (se o produto não for encontrado)
 - Corpo da Resposta (Texto):

```
Produto deletado com sucesso.
```

Exemplos de cURL

Alguns exemplos de comandos cURL para interagir com a aplicação:

Criar um Produto

```
curl -X POST http://localhost:8080/products -H "Content-Type: application/json" -d '{
  "name": "Novo Produto",
  "description": "Descrição do novo produto",
  "price": 100.0
}'
```

Listar Todos os Produtos

```
curl -X GET http://localhost:8080/products
```

Obter um Produto pelo ID

```
curl -X GET http://localhost:8080/products/{id}
```

Obs.: Substitua {id} pelo UUID do produto.

Atualizar um Produto pelo ID

```
curl -X PUT http://localhost:8080/products/{id} -H "Content-Type: application/json" -d '{
  "name": "Produto Atualizado",
  "description": "Descrição atualizada",
  "price": 150.0
}'
```

Obs.: Substitua {id} pelo UUID do produto.

Deletar um Produto pelo ID

```
curl -X DELETE http://localhost:8080/products/{id}
```

Obs.: Substitua {id} pelo UUID do produto.

Conclusão

Este guia fornece uma visão detalhada sobre como navegar e utilizar a aplicação de gestão de produtos. Seguindo os passos e formatos descritos, você deve ser capaz de interagir eficientemente com a aplicação e gerenciar os

produtos conforme necessário. Se tiver mais perguntas ou precisar de assistência adicional, sinta-se à vontade para procurar mais ajuda.