Exercícios

🚗 Desenvolvimento de API RESTful para Gerenciamento de Carros 🚀

Contexto:

Uma empresa de desenvolvimento de software decidiu criar uma solução para gerenciar um inventário de carros de forma eficiente e moderna. Utilizando Spring Boot, a meta é desenvolver uma API RESTful que permita operações básicas de CRUD sobre os carros, além de funcionalidades adicionais como atualização de placa e cálculo do valor médio dos carros por marca.

Desafio:

Implementar a API seguindo os requisitos funcionais abaixo, com ênfase na aplicação de validações específicas para o cadastro de novos carros, garantindo a unicidade e o formato correto da placa.

Requisitos Funcionais e Exemplos de URL/URI:

🆕 Cadastro de Carros:

* Método HTTP: POST
* URL Completa: <http://localhost:8080/carros>
* Validações Importantes:
  + A placa do carro deve ser única no sistema. 🚫🔁
  + A placa é obrigatória e deve ter entre 7 e 8 letras. ✔️📝
  + O ano deve ser pelo menos 1960
  + As marca e modelo são obrigatórios, mas sem mínimo de caracteres
  + O preço deve ser no mínimo 0.01
  + O campo "motivoMudancaPlaca" sempre deve ser cadastrado como nulo. Ele só poderá ser alterado no endpoint de emplacamento.
  + O "id" não é informado na requisição, pois é gerado automaticamente, conforme descrito abaixo
* Descrição: Cadastro de um novo carro, retorne o carro recém cadastrado com "id" gerado, que é um UUID. Para gerar um aleatório, basta fazer UUID.randomUUID().

📋 Listagem de Todos os Carros:

* Método HTTP: GET
* URL Completa: <http://localhost:8080/carros>
* Descrição: Retorna todos os carros cadastrados.

🔍 Consulta de Carro por ID:

* Método HTTP: GET
* URL Completa: <http://localhost:8080/carros/5052cf1c-632d-494e-a422-6cc5ebc939d2>
* Descrição: Consulta de um carro específico pelo seu ID.

🔄 Atualização de Carro:

* Método HTTP: PUT
* URL Completa: <http://localhost:8080/carros/5052cf1c-632d-494e-a422-6cc5ebc939d2>
* Descrição: Atualiza informações de um carro existente, validando novamente a placa, retornando o mesmo caso encontre.

❌ Exclusão de Carro:

* Método HTTP: DELETE
* URL Completa: <http://localhost:8080/carros/5052cf1c-632d-494e-a422-6cc5ebc939d2>
* Descrição: Remove um carro pelo seu ID, sem necessidade de retornar o objeto removido.

🆔 Atualização da Placa do Carro:

* Método HTTP: PATCH
* URL Completa: <http://localhost:8080/carros/5052cf1c-632d-494e-a422-6cc5ebc939d2/emplacamento> { "novaPlaca": "ABC1234", "motivo": "Deu ruim a anterior" }
* Descrição: Atualiza apenas a placa de um carro. Valide a placa novamente com os mesmos critérios da criação e atualização de carro. Retorna o carro no caso de sucesso.

💰 Cálculo do Valor Médio dos Carros por Marca:

* Método HTTP: GET
* URL Completa: <http://localhost:8080/carros/valor-medio?marca=Toyota>
* Descrição: Calcula o valor médio dos carros de uma marca específica.

💰 Listagem "Simples" com Cálculo do IPVA:

* Método HTTP: GET
* URL Completa: [http://localhost:8080/carros/simples-ipva](http://localhost:8080/carros/valor-medio?marca=Toyota)
* Descrição: Calcula um JSON simplificado com o valor do IPVA do carro. Traz todos os carros da API
* O IPVA é assim: É 5% do valor do veículo caso ele tenha seja do ano 2000 em diante. Caso seja anterior a 2000, o IPVA é 0.0.

Ex de JSON de cada carro:  
{  
   "marcaModelo": "Fiat Palio",  
   "ano": 2010,  
   "preco": 10000.0,  
   "IPVA": 500.0  
}

Lembrete! 🚨⚠️😱

* Não esqueça de devolver os códigos adequados para cada caso nos endpoints (sucesso, erro e etc);

Exemplos para teste:

Fusca 🤜

{  
 "id": "76321458-f6d8-47d8-9e75-3174d1448f43",  
 "marca": "Volkswagen",

"modelo": "Fusca",

"cor": "Azul",

"placa": "ANO4S56",

"preco": 30000.00,

"ano": 1972,

"motivoMudancaPlaca": ""

}

Gurgel 🇧🇷

{  
 "id": "76321458-f6d8-47d8-9e75-3174d1448f43",   
 "marca": "Gurgel",

"modelo": "X12",

"cor": "Branco",

"placa": "GUR7X12",

"preco": 15000.00,

"ano": 1980,

"motivoMudancaPlaca": ""

}

Marea 🧯🔥

{

"id": "76321458-f6d8-47d8-9e75-3174d1448f43",

"marca": "Fiat",

"modelo": "Marea",

"cor": "Vermelho incêndio",

"placa": "MRE5T98",

"preco": 20000.00,

"ano": 1999,

"motivoMudancaPlaca": ""

}

Escort 🏎️💨

{

"id": "76321458-f6d8-47d8-9e75-3174d1448f43",  
 "marca": "Ford",

"modelo": "Escort XR3",

"cor": "Azul Cobalto",

"placa": "FSC0X93",

"preco": 25000.00,

"ano": 1989,

"motivoMudancaPlaca": ""

}