

JAVA ADVANCED

A dark, moody background image showing the silhouette of a person sitting at a desk, looking at a computer monitor. The monitor displays a complex software interface with various charts and data. A bright red rectangular box is superimposed over the left side of the monitor, framing the main title text.

PROJETO MERCADO

João Carlos Lima

INTRODUÇÃO

Você é responsável por criar uma API que simula um mercado medieval, onde personagens podem cadastrar, vender, comprar e trocar itens mágicos, como espadas encantadas, elixires, grimórios, mantos de invisibilidade, etc.



PERSONAGEM

Crie o CRUD de personagem com os endpoints básicos para manipulação de dados e persistência em BD.

- nome (obrigatório)
- classe (guerreiro, mago, arqueiro)
- nível (mínimo 1, máximo 99)
- moedas (saldo para comprar itens)



ITEM

Crie os endpoints do CRUD de Item. Um item representa uma unidade que pode ser comprada no mercado do jogo.

- nome (obrigatório)
- tipo (arma, armadura, poção, acessório)
- raridade (comum, raro, épico, lendário)
- preco (valor de venda no mercado)
- dono (relacionamento com Personagem)



FILTROS E BUSCA

Usando as boas práticas de API Rest, forneça as seguintes funcionalidades:

- buscar personagem por nome
- buscar personagem por classe
- buscar itens por nome parcial
- buscar itens por tipo
- buscar itens por preço mínimo e máximo
- buscar itens por raridade



CRITÉRIOS E ENTREGA

A entrega deve ser realizada via link público do GitHub com o código do projeto na raiz do repositório. Se não atendido o projeto não será corrigido.

O link deve ser entregue até o término do prazo do Gemini. Você pode entregar o link no início da aula e ir realizando os commits até o final do prazo.

Serão avaliadas boas práticas de POO e API Rest. Os endpoints que retornam muitos dados devem ser paginados. Cada ocorrência de má prática acarretará penalidade de 5%.

O projeto pode ser realizado em dupla ou individual.