

# API Gerenciamento de Viagens e Destinos

## Descrição:

Você deve criar uma API gerenciar viagens e destinos. A API deve implementar uma relação de um para muitos (1-N) entre duas entidades: Destino e Viagem.

## Modelos:

1. Destino:
  - id: Long (identificador único do destino)
2. Viagem:
  - id: Long (identificador único da viagem)

## Requisitos da API?

1. Endpoints para Destino:
  - GET /destinos: Retorna a lista de todos os destinos.
  - GET /destinos/{id}: Retorna os detalhes de um destino específico.
  - POST /destinos: Cria um novo destino.
  - PUT /destinos/{id}: Atualiza as informações de um destino existente.
  - DELETE /destinos/{id}: Remove um destino específico.
2. Endpoints para Viagem:
  - GET /viagens: Retorna a lista de todas as viagens.
  - GET /viagens/{id}: Retorna os detalhes de uma viagem específica.
  - POST /viagens: Cria uma nova viagem. O campo "destino" deve ser preenchido com o ID do destino existente.
  - PUT /viagens/{id}: Atualiza as informações de uma viagem existente.
  - DELETE /viagens/{id}: Remove uma viagem específica.
  - GET /destinos/{destinoId}/viagens: Retorna todas as viagens associadas a um determinado destino.

## Requisitos Adicionais:

- Configure os relacionamentos entre as entidades Destino e Viagem usando anotações do JPA (@ManyToOne e @OneToMany).
- Adicione validação básica para garantir que o título da viagem, o nome do destino e o país sejam informados, e que as datas da viagem estejam em uma ordem correta (data de início antes da data de término).
- Utilize a abordagem de DTOs (Data Transfer Objects) para transferir dados

entre a API e o cliente.

- Crie mais 2 endpoints a sua escolha (comente no código qual a função dele)