**Construindo um Sistema de Blog com Node.js e Express 🚀**

Imagine que você foi contratado para desenvolver o backend de um novo sistema de blog para uma pequena empresa de conteúdo. A empresa deseja uma solução simples e eficiente, onde os usuários possam acessar uma lista de posts e visualizar o conteúdo completo de cada artigo individualmente. Você, como desenvolvedor, vai utilizar o **Express** (um framework popular do Node.js) para construir essa aplicação.

Sua tarefa é criar um **servidor web** que simula o backend desse sistema de blog. O sistema terá uma lista de posts, cada um com um título e um conteúdo. Através de rotas HTTP, os usuários poderão acessar a lista de posts e visualizar o conteúdo completo de um post específico.

**Instruções**

1. **Configuração inicial do projeto**:

* Crie uma pasta com o nome Blog e, dentro dela, inicialize um novo projeto com o nome **app.js**.
* Instale o Express.
* Instale o Nodemon.

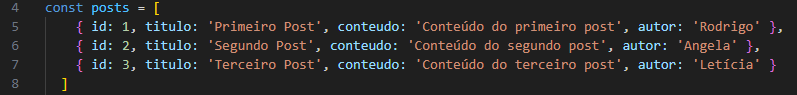
1. **Criando o servidor Express**:

* No arquivo principal **app.js**, configure um servidor Express que escute requisições na porta 3000.

1. **Definição de posts**:

* Defina um array contendo pelo menos **3 posts**. Cada post deve ter:
  + Um **ID** (número)
  + Um **título** (string)
  + Um **conteúdo** (string)
  + Um **autor** (string)

**Exemplo:**



1. **Criação das Rotas**:
   1. **Rota GET /blog**:

* Crie uma rota que retorne uma lista de todos os posts. A resposta deve conter apenas os ids e os títulos dos posts, não o conteúdo completo.
  1. **Rota GET /blog/:id**:
* Crie uma rota dinâmica que receba o id do post como parâmetro e retorne o título e o conteúdo completo do post correspondente.

1. **Testando a aplicação**:

* Acesse /blog para verificar se os títulos dos posts estão sendo retornados corretamente.
* Acesse /blog/:id (substituindo :id por um número válido, como 1, 2 ou 3) para verificar se o conteúdo completo de um post está sendo exibido.

**6. Desafios Bônus (opcional):**

**Manipulação de Erros**:

* Garanta que, ao acessar um id de post que não exista, o servidor retorne uma mensagem amigável, como "Post não encontrado".

**Método find**

* Reformule o código utilizando o método find. Dica: Peça ajuda pro Chatgpt.