***Exercícios sobre middleware***

**O QUE SÃO MIDDLEWARE:**

**Middleware é todo o tipo de função que está entre um pedido HTTP e a resposta final que o servidor envia de volta para o cliente.**

As funções de middleware podem executar as seguintes tarefas:

* Execute qualquer código.
* Faça alterações nos objetos de solicitação e resposta.
* Acabe com o ciclo de solicitação-resposta.
* Chame a próxima função de middleware na pilha.

***Funções de middleware*** são funções que têm acesso ao [objeto de solicitação](https://expressjs.com/en/4x/api.html#req) ( req), ao [objeto de resposta](https://expressjs.com/en/4x/api.html#res) ( res) e à próxima função de middleware no ciclo de solicitação-resposta do aplicativo. A próxima função de middleware é comumente denotada por uma variável chamada next.

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

<https://expressjs.com/pt-br/guide/writing-middleware.html>

<https://expressjs.com/pt-br/guide/using-middleware.html>

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

Iniciar código nodes **npm init -y**

Teste unitário Node.js

test unitario

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

npm install jest

npm install supertest

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

cria o arquivo jest.config .js

<https://jestjs.io/pt-BR/docs/getting-started>

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+++++++++++++++++++++++++++++++++++++++++++++++++++++

executar o jest no terminal : NPX jest

Git hub de test

<https://github.com/Fernanda-Kipper/email-service-backend/blob/main/src/controllers/tests/EmailController.spec.js>

Para o test do jest usarei este cara aqui