



Boas-vindas! Caro aspirante a dev!

Chegamos a nossa última aula do módulo de testes, espero que esteja ciente da importância dos testes durante um projeto de software. Vamos lá para mais uma aula?

Aula 03 – Mocks:

Esta é nossa terceira aula do módulo de Testes, nela abordaremos os seguintes tópicos:

- TDD
- Mocks

Ao final desta aula, você estará apto(a), a:

- Explicar o que são mocks e qual a sua importância; e
- Implementar mocks para testes usando a biblioteca Jest.

Exercícios:

1. O que são mocks e qual a sua importância num projeto de software?
2. É possível realizarmos testes antes de termos o código principal? Se a resposta for sim, de qual conceito estamos falando e como fazer para tal? Explique.
3. Escreva testes unitários para o **Serviço** usado na rota **POST /orders/:order_id/discount-codes** do projeto Discountopia iniciado na aula passada. O testes devem abranger as mesmas regras de negócio dessa rota já explicadas na aula passada. Como são testes unitários do Serviço, eles não devem envolver o banco de dados, nem o código de Controladores, nem o código de Repositórios. Também por serem testes do serviço, eles não devem abranger inputs inválidos que o controlador já teria descartado, como **order_id** que está ausente ou não é número. Use mocks conforme a necessidade. Seus testes devem ser escritos num arquivo `__tests__/unit/ordersService.test.js`.

Em síntese:

Ao final desta aula você saberá o que são mocks e como implementá-los em seus testes utilizando a biblioteca Jest.

Lembre-se que o foco deste estudo é sempre aprofundar o seu conhecimento em web e, portanto, não se limite às referências deste módulo e mergulhe no conhecimento sobre Testes, pertinentes a esta aula.



Referências:

- A. [O que são técnicas de mock em testes, e quais os principais tipos? - DEV Community.](#)
- B. [Conceitos de Mocks e Stubs em Teste de Unidade](#)
- C. [Testes Unitários 101: Mocks, Stubs, Spies e todas essas palavras difíceis | by Lucas Santos | Training Center | Medium](#)
- D. [Simulações Manuais · Jest](#)
- E. [Mock Functions · Jest.](#)
- F. Uma história de testes com Jest (básico): <https://alluring-beluga-3dd.notion.site/Uma-hist-ria-de-testes-com-Jest-b-sico-6ef0d86483fb4514851f1397c4a6b3cb>
 - a. Veja a **Parte 3**