

# 4 - Validando o preenchimento das informações de um input

**Descrição da tarefa:** Nessa tarefa, você deve criar um ou mais inputs na tela. Ao digitar qualquer texto em um desses inputs, a sua borda deverá ficar com a cor verde. Caso o input estiver vazio, a cor da borda deverá voltar ao normal.

💡 **Dica:** É essencial começar pensando no objetivo do que precisa ser feito e no passo a passo para chegar lá.

Evite utilizar o style do JavaScript para esses exercícios

NÃO SAIA TENTANDO ESCREVER CÓDIGO NO JS! Antes disso você precisa saber O QUE precisa ser feito e só depois COMO fazer. 🙏

Inputs sem nenhuma informação:

Two empty text input fields are shown side-by-side within a white rectangular container. Both fields have a thin black border and no text inside.

Como devem ficar ao preencher:

Two text input fields are shown side-by-side within a white rectangular container. The first field on the left contains the text 'escrevi' and has a thick green border. The second field on the right is empty and has a thin black border.

▼ Resolução 00 (só olhe depois de fazer o exercício)

✓ HTML:

```
<!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta
http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport"
content="width=device-width, initial-scale=1.0"> <title>Exercício 4 -
JS Intermediário</title> <link rel="stylesheet" href="./style.css">
</head> <body> <input type="text" class="input-text"> <input
type="text" class="input-text"> <script src="./index.js"></script>
</body> </html>
```

✓ CSS:

```
.campo-preenchido{ border: 1px solid green; }
```

✓ JS:

```
/* Dica: Começar pensando no objetivo do que precisa fazer e no passo a passo pra chegar lá é essencial. NÃO SAIA TENTANDO ESCREVER CÓDIGO! Antes disso você precisa saber O QUE precisa ser feito e só depois COMO fazer. Objetivo do exercício: Ao preencher alguma informação nos inputs, deve ser verificado se estão vazios ou não. Caso tiver algum valor, será adicionada a borda verde. Se não tiver, é removida a borda. Passo 1 - Pegar todos os inputs que tenham a mesma classe. Passo 2 - Dar um jeito de percorrer cada input. Passo 3 - Quando o usuário preencher algo, precisamos disparar a ação para verificar se o input foi alterado. Passo 4 - Realizar a verificação se o valor do
```