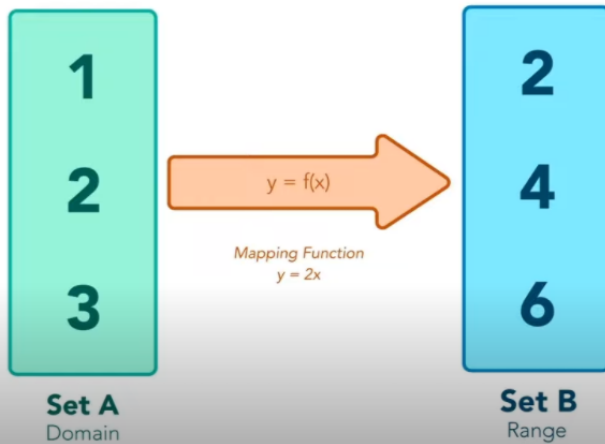


O que é



- Cria um novo array
- Não modifica o array original
- Realiza as operações em ordem

Maps é quando um conjunto passa por uma operação e se torna outro conjunto.

Os números do conjunto A passaram por uma função chamada Mapping Function onde $Y = 2x$, logo o primeiro elemento do conjunto B vai ser 2, assim como no lugar de 2 vai ser o 4 e no lugar de 3 vai ser o 6.

Aula 1 | Etapa 2:

Sintaxe

Map, filter e reduce

Sintaxe



```
array.map(callback, thisArg)
```



```
callback(item, index, array)
```

Implementando o método no código.

Utilizamos o [array.map](#) e receber dois itens que são o callback e o thisArg

Callback: função a ser executada em cada elemento

Dentro do callback também temos acesso tanto ao item que você deseja modificar como o index desse item e ao array original

thisArg (opcional): valor de 'this' dentro da função de callback

O argumento 'this' é opcional apenas se você quiser fazer uma operação que vai variar de acordo com o objeto que você precisa que essa operação seja feita dentro desse objeto

Map vs forEach

Map, filter e reduce

O for Each vai fazer uma operação em cada elemento desse array

```
// Usando map
const array = [1, 2, 3, 4, 5];
array.map((item) => item * 2); // retorno: [2, 4, 6, 8, 10];

// Usando forEach
const array = [1, 2, 3, 4, 5];
array.forEach((item) => item * 2); // retorno: undefined
```

Para que o resultado do forEach apareça é necessário que ele esteja dentro de alguma constante

- Valor de retorno
- Considere se o array auxiliar será necessário