

## Para saber mais: filtrando sem regex?



87%

ATIVIDADES  
8 DE 8FÓRUM DO  
CURSOVOLTAR  
PARA  
DASHBOARDMODO  
NOTURNOABRIR  
CADERNO

59.7k xp

Aprendemos como filtrar, comparar letra a letra, utilizando expressões regulares, representado pelo fragmento de código abaixo:

```
var expressao = new RegExp(this.value,"i");  
if (!expressao.test(nome)) {  
    paciente.classList.add("invisivel");  
} else{  
    paciente.classList.remove("invisivel");  
}
```

[COPIAR CÓDIGO](#)

Mas há um modo de fazer essa comparação sem a necessidade de utilizar expressões regulares! Podemos utilizar a função `substring`, que recebe dois parâmetros, fazendo com que ela devolva parte da string, com o tamanho definido nos parâmetros. O primeiro parâmetro é o início, começando do 0 (que representa o primeiro caractere). O segundo parâmetro define o fim (exclusivo, mostramos até o penúltimo caractere). Por exemplo:



87%

ATIVIDADES  
8 DE 8

FÓRUM DO  
CURSO

VOLTAR  
PARA  
DASHBOARD

MODO  
NOTURNO

ABRIR  
CADERNO



59.7k xp

```
var string = "Alura";  
var resultado = string.substring(1, 4);
```

COPIAR CÓDIGO

Extraímos uma string que começa no segundo caractere (número 1) e termina no quarto caractere (número 3, índice anterior ao número 4). O valor da variável `resultado` é:

lur

COPIAR CÓDIGO

Conhecendo essa função, pense em um modo de comparar o valor digitado com parte do nome, sem utilizar expressões regulares. Veja em seguida a resposta do instrutor.

## Opinião do instrutor



Como o primeiro parâmetro é o início, e queremos comparar desde o início da string `nome`, vamos utilizar como início o valor `0`, ou seja,



87%

ATIVIDADES  
8 DE 8

FÓRUM DO  
CURSO

VOLTAR  
PARA  
DASHBOARD

MODO  
NOTURNO

ABRIR  
CADERNO



59.7k xp

sempre a partir do primeiro caractere. Mas qual é o fim? O fim é justamente o tamanho do valor digitado:

```
nome.substr(0, this.value.length);
```

COPIAR CÓDIGO

Podemos guardar essa string em uma variável, e compará-la com o que está sendo digitado. Caso seja falso, adicionamos a classe `invisivel`, se não for, removemos-a:

```
var comparavel = nome.substr(0, this.value.length);  
if (!(this.value == comparavel)) {  
    paciente.classList.add("invisivel");  
} else{  
    paciente.classList.remove("invisivel");  
}
```

COPIAR CÓDIGO

Mas e a distinção entre letras maiúsculas e minúsculas? Nesse caso não temos distinção entre letras maiúsculas e minúsculas, mas para contornar isso, antes de compará-las, podemos colocar as duas strings em letras minúsculas, para efetuar a comparação entre elas em seguida:



87%

ATIVIDADES  
8 DE 8

FÓRUM DO  
CURSO

VOLTAR  
PARA  
DASHBOARD

MODOS  
NOTURNO

ABRIR  
CADERNO



59.7k xp

```
var comparavel = nome.substr(0, this.value.length);
var comparavelMinusculo = comparavel.toLowerCase();
var valorDigitadoMinusculo =
this.value.toLowerCase();

if (!(valorDigitadoMinusculo == comparavelMinusculo))
{
    paciente.classList.add("invisivel");
} else{
    paciente.classList.remove("invisivel");
}
```

COPIAR CÓDIGO

Esta é uma alternativa de implementar a mesma funcionalidade sem expressão regular, porém temos que escrever mais e nos preocupar com mais detalhes! Fica ai esta opção para você guardar nos seus conhecimentos.