

REGULAR EXPRESSION

Regexp Substituição

Referência da Seleção

É possível utilizarmos o `$&` durante o momento da substituição para fazermos uma referência à seleção.

```
// Procura: Java  
const regexp = /Java/g;  
  
'PHP e Java são linguagens diferentes'.replace(regexp, '-  
-$&Script');  
// PHP e --JavaScript são linguagens diferentes  
// $& será igual à Java
```

Grupo de Captura

É possível definirmos diferentes grupos de captura, que poderão ser referenciados durante a substituição. Basta envolvermos um grupo entre `()` parênteses. A referência se cada grupo será feita com `$n`, sendo o primeiro `$1`.

```
// Procura: sequência alfanumérica, seguida  
// de @, seguido de alfanumérico ou .  
const regexp = /(\w+)@[\w.]+/g;  
  
'andre@email.com.br'.replace(regexp, '$1@gmail.com');  
// andre@gmail.com
```

*Não use este regexp para emails,
ele falha em alguns casos.*

Mais de um Grupo

Podemos definir quantos grupos de captura quisermos.

```
// Procura: sequência alfanumérica, seguida  
// de , seguido espaço de sequência alfanumérica.  
const regexp = /(\w+),\s(\w+)/g;  
  
'Rafael, Andre'.replace(regexp, '$2 $1');  
// Andre Rafael
```

Mais do que Captura apenas

Um grupo também serve para agruparmos uma sequência de caracteres que queremos em repetição.

```
// Procura: qualquer sequência de ta  
const regexp = /(ta)+/gi;  
  
'Tatata, tata, ta'.replace(regexp, 'Pá');  
// Pá, Pá, Pá
```

Ignorar Captura

Utilize o `(?:)` para ignorar a captura.

```
// Procura: qualquer sequência de ta  
const regexp = /(?:ta)+/gi;  
  
'Tatata, tata, ta'.replace(regexp, 'Pá');  
// Pá, Pá, Pá
```

Positive Lookahead

Faz a seleção dos itens que possuírem o padrão dentro de `(?=)` à sua frente. Apesar de utilizar `()` parênteses o positive lookahead não captura grupo.

```
// Procura: dígitos em sequência, que  
// possuírem px, sem selecionar o px.  
const regexp = /\d(=px)/g;  
  
'2em, 4px, 5%, 2px, 1px'.replace(regexp, 'X');  
// 2em, Xpx, 5%, Xpx, Xpx
```

Negative Lookahead

Faz a seleção dos itens não possuírem o padrão dentro de `(?!)` à sua frente.

```
// Procura: dígitos que não possuïrem px
// sem selecionar o restante.
const regexp = /\d(?!px)/g;

'2em, 4px, 5%, 5px, 1px'.replace(regexp, 'X');
// Xem, 4px, X%, 5px, 1px
```


Positive Lookbehind

Faz a seleção dos itens que possuírem o padrão dentro de `(?<=)` atrás dos mesmos.

```
// Procura: dígitos que possuírem R$  
// na frente dos mesmos  
const regexp = /(?<=R\$)[\d]+/g;  
  
'R$99, 100, 200, R$20'.replace(regexp, 'X');  
// R$X, 100, 200, R$X
```