

05

## Para saber mais : Todo que não é!

Na aula vimos como selecionar uma tag específica dentro de um alvo e chegamos juntos a regex em baixo:

```
<h1.+?>([\w\sõäí.]+)</h1>
```

Com ela pegamos a tag `<h1>`, e o grupo devolve o conteúdo da tag. Repare também que deixamos o *quantifier preguiçoso* para não pegar a tag inteira.

Por exemplo a regex `<h1.+>` seleciona o tag inteira:

Target string (alvo)

```
<h1 class="text-left">Expressões regulares</h1>
```

Pattern (expressão regular)

```
<h1.+>
```

**Executar Regex**

Mostra índice    Mostra grupos

**1 Matches (resultados)**

```
<h1 class="text-left">Expressões regulares</h1>
```

**Highlight**

```
<h1 class="text-left">Expressões regulares</h1>
```

Mas se usarmos o quantifier preguiçosamente selecionamos apenas a abertura da tag:

```
<h1.+?>
```

Target string (alvo)

```
<h1 class="text-left">Expressões regulares</h1>
```

Pattern (expressão regular)

```
<h1.+?>
```

**Executar Regex**

Mostra índice    Mostra grupos

**1 Matches (resultados)**

```
<h1 class="text-left">
```

**Highlight**

```
<h1 class="text-left">Expressões regulares</h1>
```

Há uma alternativa para resolver esse problema de abertura da tag. Podemos definir uma classe de caracteres que seleciona *tudo que não é um >*. Essa negação é feita através da meta-char `^`.

Teste o exemplo abaixo sem usar um quantifier preguiçoso:

<h1[^>]+>

**Target string (alvo)**

```
<h1 class="text-left">Expressões regulares</h1>
```

**Pattern (expressão regular)**

```
<h1[^>]+>
```

**Executar Regex**

Mostra índice     Mostra grupos

**1 Matches (resultados)**

```
<h1 class="text-left">
```

**Highlight**

**<h1 class="text-left">Expressões regulares</h1>**

Repare que usamos a classe em conjunto com o quantifier *ganancioso*: [^>]+

Mesmo assim foi encontrado apenas a abertura da tag pois a nossa classe de caractere **não inclui** a caractere >