

01

## Selecionando com âncoras

### Transcrição

Vamos continuar com nosso exemplo CSV e copiar a segunda linha para nosso formulário:

```
denise teste, 987.654.321-00,28 de Dezembro de 1991,(31)45562712,SCS Qd. 8 Bl. B-50,11,70333-900,Ric
```

Um problema comum para resolver no dia a dia é a procura de uma palavra específica. Nesse exemplo, vamos tentar achar a preposição `de`. A primeira ideia seria colocar o valor literal `de`. Ao testar, percebemos que a regex pega mais do que o desejado, como `denise` ou `Grande`. Com aquilo que já aprendemos, vamos refinar a regex e colocar um `whitespace` no início e no fim da preposição: `\sde\s`

Agora sim, a regex só encontra as palavras corretas, no entanto também selecionou o *espaço no início e fim* da preposição.

### Word boundary

O que queremos na verdade é encontrar o início e o fim de uma palavra, que podemos definir pela regex `\b`. `\b` é uma das **âncoras**, chamada de **word boundary**. Uma âncora não seleciona um caractere, e sim apenas uma posição no texto alvo. Ou seja, a âncora `\b` sozinha devolve nada (*zero-length*) mas combinada com um valor literal, ou outra regex, podemos finalmente definir como encontrar a preposição `de`:

```
\bde\b
```

Repare que usamos o `\b` no início e no fim, ou seja, queremos buscar exatamente a preposição `de`. Testando isso no nosso formulário, agora sim só selecionamos apenas a palavra `de`. Vamos testar a âncora em um outro exemplo para deixar bem claro como ela funciona. Abaixo temos uma sequência de letras `a` e queremos selecionar as sequências com três `a` (`a{3}`):

```
aaa aaaa aaa aaaa aaa
```

Vamos testar uma vez sem `\b`, uma vez com `\b` no início, e outra com `\b` no início e fim:

```
a{3}
```

```
aaa aaaa aaa aaaa aaa
```

```
a{3}\b
```

```
aaa aaaa aaa aaaa aaa
```

\ba{3}\b

aaa aaaa aaa aaaa aaa