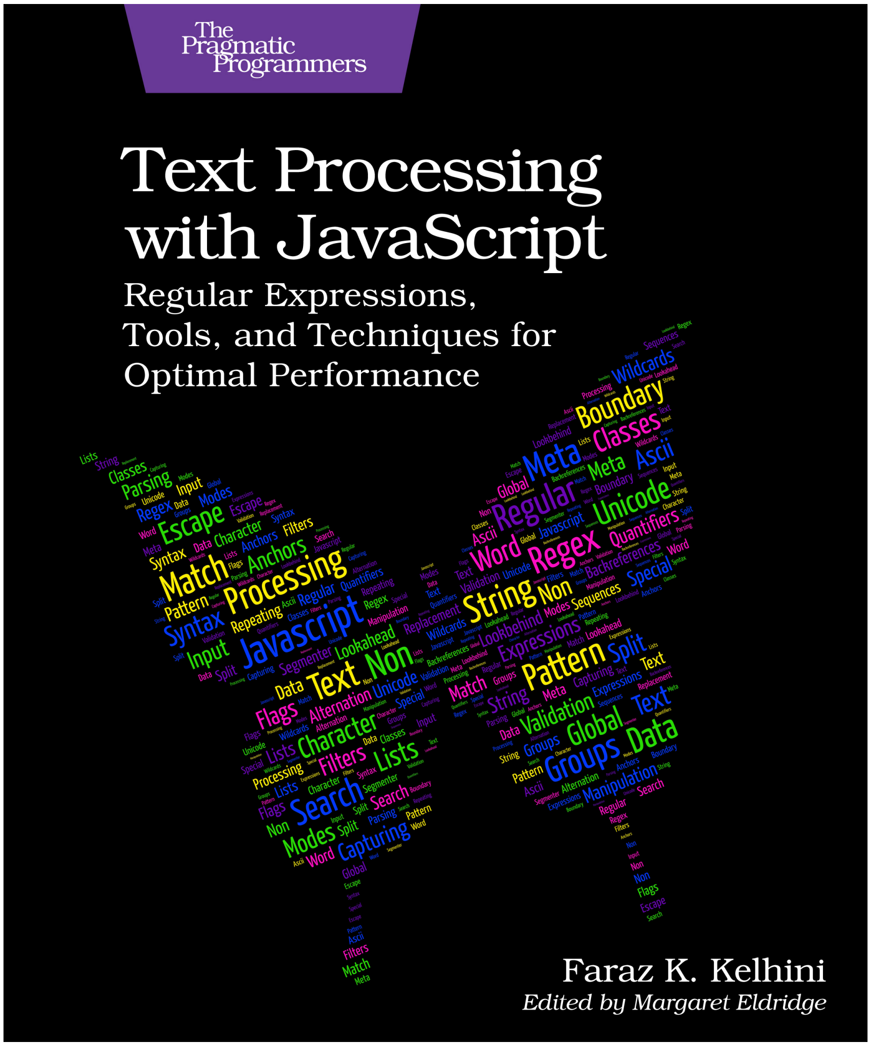

The Pragmatic Programmers

Text Processing with JavaScript

Regular Expressions, Tools, and Techniques for Optimal Performance

Faraz K. Kelhini

Edited by Margaret Eldridge

Recipe 27

Matching One of Several Characters with the Character Class

Task

Suppose you want to find a word in a document even if it is misspelled. For example, the word "license" is one of the most misspelled words in English. You want to write a pattern that matches "license," "licence," "lisence," or "lisense" in a document.

Solution

Use a character class:

part_2/character_class/character_class_ex1.js
```
const re = /li[sc]en[sc]e/;

re.test("A driver's license");    // → true
re.test("A driver's lisense");    // → true
re.test("A driver's licence");    // → true
re.test("A shopping list");       // → false
```

A character class matches only one out of the specified characters. In this code, the specified characters are "s" and "c," so the regex matches "license," "licence," "lisense," or "lisence," but not "liscense." Keep in mind that a character class matches only one character.

Discussion

Certain characters change the behavior of the character class. If you place a caret (^) after the opening square bracket, it negates the entire character class. That means the character class will match any character that isn't one of the specified characters.

So, /[^license]/ would match any character that isn't "l," "i," "c," "n," "s," and "e":

part_2/character_class/character_class_ex2.js
```
const re = /[^license]/;

re.test("lie");     // → false
re.test("list");    // → true
```

This pattern matches “t” in “list,” so test() returns true. You might have noticed that the caret (^) is the same as the one that matches the beginning of a string (Recipe 24, Asserting the Start or End of a String with ^ and $, on page ?). Although the character is the same, its meaning is entirely different.

It’s just like the English word “arm” can mean different things depending on what context it is used in (sometimes a part of the body, sometimes to provide a weapon).

Also, keep in mind that caret has a special meaning only when used right after the opening bracket of the character class. So, /[a^bc]/ wouldn’t negate the character class because “^” would be treated as a literal character.

As with a regular class, a negated class must match a character to be successful. For example, the pattern /Number[^5]/ matches “Number6” but not “Number” because the class expects a character:

part_2/character_class/character_class_ex3.js

```
const re = /Number[^5]/;

re.test("Number6");    // → true
re.test("Number");     // → false
```

Use a character class to match a character out of several characters, such as when you want to consider misspelled words or spelling differences in American and British English.

Use a negated character class to list characters you don’t want to appear in a string. An exciting aspect of the character class is its ability to match a range of characters, which you’ll learn about in the next recipe.