



🏠 → [Regular expressions](#)

📅 4th September 2019

Anchors: string start ^ and end \$

The caret ^ and dollar \$ characters have special meaning in a regexp. They are called “anchors”.

The caret ^ matches at the beginning of the text, and the dollar \$ – at the end.

For instance, let's test if the text starts with `Mary` :

```
1 let str1 = "Mary had a little lamb";
2 alert( /^Mary/.test(str1) ); // true
```



The pattern ^Mary means: “string start and then Mary”.

Similar to this, we can test if the string ends with `snow` using snow\$:

```
1 let str1 = "it's fleece was white as snow";
2 alert( /snow$/.test(str1) ); // true
```



In these particular cases we could use string methods `startsWith/endsWith` instead. Regular expressions should be used for more complex tests.

Testing for a full match

Both anchors together ^...\$ are often used to test whether or not a string fully matches the pattern. For instance, to check if the user input is in the right format.

Let's check whether or not a string is a time in `12:34` format. That is: two digits, then a colon, and then another two digits.

In regular expressions language that's \d\d:\d\d :

```
1 let goodInput = "12:34";
2 let badInput = "12:345";
3
4 let regexp = /^\\d\\d:\\d\\d$/;
5 alert( regexp.test(goodInput) ); // true
6 alert( regexp.test(badInput) ); // false
```



Here the match for \\d\\d:\\d\\d must start exactly after the beginning of the text ^, and the end \$ must immediately follow.

The whole string must be exactly in this format. If there's any deviation or an extra character, the result is `false`.

Anchors behave differently if flag `m` is present. We'll see that in the next article.

Anchors have “zero width”

Anchors `^` and `$` are tests. They have zero width.

In other words, they do not match a character, but rather force the regexp engine to check the condition (text start/end).

Tasks

Regex `^$`

Which string matches the pattern `^$` ?

solution



Previous lesson

Next lesson



Share



[Tutorial map](#)

Comments

- If you have suggestions what to improve - please [submit a GitHub issue](#) or a pull request instead of commenting.
- If you can't understand something in the article – please elaborate.
- To insert a few words of code, use the `<code>` tag, for several lines – use `<pre>`, for more than 10 lines – use a sandbox ([plnkr](#), [JSBin](#), [codepen](#)...)