

Раздел


[Регулярные выражения](#)

Навигация по уроку

Обратная ссылка по номеру:  
`\N`Обратная ссылка по имени:  
`\k<имя>`

Комментарии

Поделиться

[Редактировать на GitHub](#) → [Регулярные выражения](#) 6-го сентября 2019

## Обратные ссылки в шаблоне: `\N` и `\k<имя>`

Доступ к содержимому скобочных групп (`(...)`) есть не только в результате поиска и при замене, но и в самом шаблоне.

### Обратная ссылка по номеру: `\N`

К группе можно обратиться в шаблоне, используя `\N`, где `N` – это номер группы.

Чтобы было яснее, зачем это нужно, рассмотрим пример.

Необходимо найти строки в кавычках: либо одинарных `'...'`, либо двойных `"..."` – оба варианта должны подходить.

Как найти такие строки?

Можно попытаться добавить оба вида кавычек в квадратные скобки: `['"](.*)['"]`, но в таком случае будут находиться строки со смешанными кавычками, например `"...'"` и `'..."`. Это приведёт к ошибке, когда одна кавычка окажется внутри других, как в строке `"She's the one!"`:

```
1 let str = `He said: "She's the one!".`;
2
3 let regexp = /[["](.*)[""]]/g;
4
5 // Результат - не тот, который хотелось бы
6 alert( str.match(regexp) ); // "She"
```

Как видно, шаблон нашёл открывающую кавычку `"`, а после нашёл текст вплоть до следующей кавычки `'`, после чего поиск завершился.

Для того, чтобы шаблон искал закрывающую кавычку такую же, как и открывающую, обернём открывающие кавычки в скобочную группу и используем обратную ссылку на неё: `([""])(.*)\1`.

Вот верный код:

```
1 let str = `He said: "She's the one!".`;
2
3 let regexp = /[(")](.*)\1/g;
4
5 alert( str.match(regexp) ); // "She's the one!"
```

Теперь работает! Движок регулярных выражений находит первую кавычку из шаблона `([""])` и запоминает её содержимое. Это первая скобочная группа.

Далее в шаблоне `\1` означает «найти то же самое, что в первой скобочной группе», а именно – аналогичную кавычку в нашем случае.

Аналогично, `\2` означает содержимое второй скобочной группы, `\3` – третьей, и так далее.

#### На заметку:

Мы не можем обратиться к группе, которая исключена из запоминания при помощи `?:`.

Раздел

[Регулярные выражения](#)

Навигация по уроку

Обратная ссылка по номеру:  
`\N`

Обратная ссылка по имени:  
`\k<имя>`

Комментарии

Поделиться



[Редактировать на GitHub](#)

 **Не перепутайте:** в шаблоне `\1`, при замене `$1`

В строке замены для вставки группы мы используем доллар: `$1`, а в шаблоне обратный слеш `\1`.



## Обратная ссылка по имени: `\k<имя>`

Если в регулярном выражении много скобочных групп, то удобно давать им имена.

Для обращения к именованной группе можно использовать синтаксис `\k<имя>`.

В примере ниже кавычки обозначены `?<quote>`, так что обращение будет `\k<quote>`:

```
1 let str = `He said: "She's the one!".`;
2
3 let regexp = /(?<quote>['"])(.*?)\k<quote>/g;
4
5 alert( str.match(regexp) ); // "She's the one!"
```



Проводим [курсы по JavaScript и фреймворкам](#).



## Комментарии

перед тем как писать...

