Регулярные выражения в JavaScript являются объектами класса [RegExp](https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/RegExp).

Кроме того, методы для поиска по регулярным выражениям встроены прямо в обычные строки String.

**[str.search(reg)](https://learn.javascript.ru/regexp-methods" \l "str-search-reg)**

## [str.match(reg) без флага g](https://learn.javascript.ru/regexp-methods" \l "str-match-reg-bez-flaga-g) - если регэксп без флага g – ищет совпадение с подрезультатами в скобках

## [str.match(reg) с флагом g](https://learn.javascript.ru/regexp-methods" \l "str-match-reg-s-flagom-g) -  если регэксп с флагом g – ищет все совпадения, но без скобочных групп.

## [str.split(reg|substr, limit)](https://learn.javascript.ru/regexp-methods" \l "str-split-reg-substr-limit)

## [str.replace(reg, str|func)](https://learn.javascript.ru/regexp-methods" \l "str-replace-reg-str-func)

## методы класса RegExp

Методы становятся гораздо понятнее, если разбить их использование по задачам, которые нужны в реальной жизни.

**Для поиска только одного совпадения:**

* Найти позицию первого совпадения – str.search(reg).
* Найти само совпадение – str.match(reg).
* Проверить, есть ли хоть одно совпадение – regexp.test(str) или str.search(reg) != -1.
* Найти совпадение с нужной позиции – regexp.exec(str), начальную позицию поиска задать в regexp.lastIndex.

**Для поиска всех совпадений:**

* Найти массив совпадений – str.match(reg), с флагом g.
* Получить все совпадения, с подробной информацией о каждом – regexp.exec(str) с флагом g, в цикле.

**Для поиска-и-замены:**

* Замена на другую строку или результат функции – str.replace(reg, str|func)

**Для разбивки строки на части:**

* str.split(str|reg)

Зная эти методы, мы уже можем использовать регулярные выражения.

Конечно, для этого желательно хорошо понимать их синтаксис и возможности, так что переходим к ним дальше.

## [Важнейшие классы: \d \s \w](https://learn.javascript.ru/regexp-character-classes" \l "vazhneyshie-klassy-d-s-w)

Мы рассмотрели классы для поиска типов символов:

* \d – цифры.
* \D – не-цифры.
* \s – пробельные символы, переводы строки.
* \S – всё, кроме \s.
* \w – латинница, цифры, подчёркивание '\_'.
* \W – всё, кроме \w.
* '.' – точка обозначает любой символ, кроме перевода строки.

Если хочется поискать именно сочетание "\d" или символ «точка», то его экранируют обратным слэшем, вот так: \.

Заметим, что регулярное выражение может также содержать перевод строки \n, табуляцию \t и прочие спецсимволы для строк. Конфликта с классами не происходит, так как для них зарезервированы другие буквы.