

Регулярные выражения Python

это шаблоны для поиска, описываемые с помощью специального синтаксиса.

```
import re
```

Для экранирования служебных символов в шаблонах поиска и замены используют два способа:

- обратный слэш `\`
- «сырые» строки `r''`

Функции Regex

- `re.match(r'шаблон', строка)` - находит вхождение фрагмента в начале строки
 - `re.search()` - находит первое вхождение фрагмента в любом месте и возвращает объект `match`
 - `re.span()` - возвращает кортеж, содержащий начальную и конечную позиции искомого фрагмента
 - `re.string()` - вернет строку, переданную в функцию `re.search`
 - `re.group()` - возвращает фрагмент строки, в котором было обнаружено совпадение
 - `re.findall()` - находит все вхождения фрагмента
 - `re.split()` - расщепляет строку по заданному шаблону
 - `re.sub()` - заменяет фрагмент в соответствии с шаблоном
 - `re.compile()` - создает объект из регулярного выражения
-
- `.` - используется для поиска соответствия любому единичному символу, кроме символа новой строки
 - `^` - выделяет начало строки
 - `$` - выделяет конец строки
 - `*` - используется для 0 или более значений крайнего левого символа / «жадные» операторы
 - `+` - используется для 1 или более значений крайнего левого символа / «жадные» операторы
 - `?` - от нуля до одного вхождения, то же самое, что `{0,1}` / «ленивые» операторы
 - `{}` - указывает количество вхождений, можно задавать единичным числом или диапазоном / «жадные» операторы
 - `[]` - ищем любой символ в квадратных скобках
 - `\` - экранирование
 - `|` - логическое ИЛИ
 - `()` - группировка
 - `[^...]` - любой символ, кроме указанных в скобках
-
- `\w` - соответствие любой букве, цифре или подчеркиванию, эквивалентен `[a-zA-Z0-9_]`

- `\w` - соответствие любому символу, кроме буквенного и цифрового символа и знака подчёркивания
 - `\d` - соответствие любому цифровому символу, эквивалентен `[0–9]`
 - `\D` - соответствие любому нецифровому символу
 - `\s` - любой пробельный символ (пробел, новая строка, табуляция, возврат каретки и тому подобное)
 - `\S` - любой символ, кроме пробельного
 - `\A` - начало строки, то же самое, что `^`
 - `\Z` - конец строки, то же самое, что `$`
 - `\b` - начало или конец слова
 - `\B` - середина слова
 - `\n, \t, \r` - новая строка, табуляция, возврат каретки
-

Флаги

- `g` (global) - не возвращает результат после первого совпадения, а продолжает поиск с конца предыдущего совпадения
 - `m` (multi line) - с таким флагом, операторы `^` и `$` вызовут совпадение в начале и конце строки ввода (line), вместо строки целиком (string)
 - `i` (insensitive) - делает выражение регистронезависимым (например, `/aBc/i` соответствует `AbC`)
-

<https://regex101.com> - сайт, позволяющий тестировать регулярные выражения

<https://docs.python.org/3/library/re.html> - оригинальная документация