

## Регулярные выражения

1. Дана строка `'x1x x2x x3x x4x x5x xb x xv x'`.  
Написать регулярное выражение, ищущее в этой строке подстроки, по краям которых стоят буквы `'x'`, а между буквами `'x'` – одна цифра.
2. Дана строка `'x1x x22x x333x x4444x x55555x xx xc x xtpx'`.  
Написать регулярное выражение, ищущее в этой строке подстроки, в которых по краям стоят буквы `'x'`, а между буквами `'x'` – любое количество цифр (включая ноль цифр – `'xx'`).
3. Дана строка `'aba aea aca aza axa'`.  
Написать регулярное выражение, которое в этой строке найдет только следующие подстроки: `'aba'`, `'aea'`, `'axa'`.
4. Дана строка `'aba aea aca aza axa a.a a+a a*a'`.  
Написать регулярное выражение, которое в этой строке найдет только следующие подстроки: `'aba'`, `'a.a'`, `'a+a'`, `'a*a'`.
5. Дана строка `'aaa aBa aca aDa afa aZa aga'`.  
Написать регулярное выражение, ищущее в этой строке подстроки, по краям которых стоят буквы `'a'`, а между буквами `'a'` – буква из набора от `a` до `f` и от `A` до `Z`.
6. Написать регулярное выражение, которое с помощью метода `test` проверяет, является ли строка URL-адресом страницы, оканчивающимся на `.com` или `.ru`, протокол может быть как `http`, так и `https`.

Регулярное выражение должно корректно работать для URL-адресов следующего вида:

1) `https://www.yandex.ru` 2) `http://www.yandex.ru`

3) `https://www.google.com`

7. С использованием регулярного выражения написать функцию, возвращающую сумму чисел, содержащихся в строке.

Пример работы: `func("Hello World! 1 23 JavaScript 456") // 480`

8. С использованием регулярного выражения написать функцию, возвращающую количество слов в предложении. Слова могут быть отделены друг от друга более, чем одним пробелом.

Пример работы: `func('Oh, hey there! What is going on? Tell me.') // 9`