

Регулярные выражения

Регулярное выражение (или RE) определяет набор строк, который ему соответствует; функции в этом задании позволяют проверить, соответствует ли конкретная строка заданному регулярному выражению.

Сведения по использованию регулярных выражений в Питоне можно найти в Интернете, в частности на сайтах:

1. [Регулярные выражения в Python от простого к сложному. Подробности, примеры, картинки, упражнения / Хабр \(habr.com\)](#)
2. [Регулярные выражения в Python: теория и практика \(tproger.ru\)](#)
3. [Регулярные выражения в Python: подробная инструкция по применению / Skillbox Media](#)

Задачи на регулярные выражения

1. Напишите программу на Python,
 - a. чтобы проверить, что строка содержит только определенный набор символов (в данном случае a-z, A-Z и 0-9).
 - b. чтобы найти последовательности строчных букв, соединенные подчеркиванием.
 - c. чтобы найти последовательности одной буквы верхнего регистра, за которой следуют буквы нижнего регистра.
 - d. чтобы она находила строки, содержащие только прописные и строчные буквы, цифры и подчеркивания.
2. Запишите программу на Python, которая выделяет строку, имеющую символ «a»,
 - a. за которым следует ноль или более «b»;
 - b. за которым следует один или несколько «b»;
 - c. за которым следует ноль или один «b»;
 - d. за которым следует три «b». Напишите программу на Python, которая соответствует строке;
 - e. за которой следует что-либо, заканчивающееся на «b»;
 - f. за которым следует от двух до трех «b».
3. Напишите программу на Python, которая
 - a. выделяет слово в начале строки.
 - b. выделяет слово в конце строки, с необязательной пунктуацией.
 - c. выделяет слово, содержащее «z».
 - d. выделяет слово, содержащее 'z', но не в начале или конце слова.
4. Напишите программу на Python,
 - a. чтобы удалить ведущие нули из чисел IP-адреса.
 - b. чтобы проверить наличие заданной цифры в конце строки.
 - c. для поиска чисел (0-9) длиной от 1 до 3 в данной строке.
5. Напишите программу на языке Python для поиска определенных подстрок в строках.
Пример текста: «Fast brown fox leaps over lazy dog».
Искомые слова: 'fox', 'dog', 'horse'

6. Написать программу на Python

- a. для поиска литерной строки в строке, а также места в оригинальной строке, где встречается шаблон.
Примерный текст : 'The quick brown fox jumps over the lazy dog.'
Искомое слово : 'fox'
- b. для извлечения значений между кавычками строки.
- c. для выделения и печати чисел в данной строке.

7. Напишите программу на Python,

- a. чтобы найти вхождение и место подстрок в строке;
- b. чтобы выделить и напечатать числа и их положение в данной строке.
- c. чтобы найти все пятисимвольные слова в строке.
- d. чтобы найти все слова из трех, четырех и пяти символов в строке.
- e. чтобы найти все слова длиной не менее 4 символов в строке.

8. Напишите программу на Python,

- a. чтобы заменить пробелы подчеркиванием и наоборот.
- b. чтобы сократить 'Road' на 'Rd.' в данной строке.
- c. чтобы заменить все вхождения пробела, запятой или точки двоеточием.
- d. чтобы заменить максимум 2 вхождения пробела, запятой или точки на двоеточие.

9. Напишите программу на Python,

- a. чтобы извлечь год, месяц и дату из URL.

Пример URL:

"https://www.washingtonpost.com/news/football-insider/wp/2016/09/02/odell-beckhams-fame-rests-on-one-stupid-little-ball-josh-norman-tells-author/"

- b. чтобы преобразовать дату в формате гggг-мм-дд в формат дд-мм-гггг.

10. Напишите программу на Python,

- a. чтобы установить, есть ли два слова из списка слов, начинающихся с буквы «Р».
- b. чтобы найти все слова, начинающиеся с «а» или «е» в данной строке.
- c. чтобы проверить, является ли строка десятичным числом с точностью до 2 знаков после запятой.

11. Напишите программу на Python

- a. для преобразования строки camel-case в строку snake-case.
- b. для преобразования строки snake-case в строку camel-case.
- c. чтобы преобразовать данную строку в snake-case строку.

Пример выходных данных:

btw... -what- * -do * -you-call-that-naming-style? -snake-case?

12. Напишите программу на Python,

- a. чтобы удалить множественные пробелы из строки.
- b. чтобы удалить из строки все, кроме буквенно-цифровых символов.
- c. чтобы удалить из строки слова длиной от 1 до заданного числа.
- d. для удаления подстрок нижнего регистра из данной строки.

13. Напишите программу на Python,

- a. чтобы найти URL-адреса в строке.

- b. чтобы найти все английские наречия и их позиции в данном предложении.
Образец текста: "Clearly, he has no hardly excuse for such behavior."

14. Напишите программу на Python,

- a. чтобы разделить строку по прописным буквам.
- b. чтобы выполнить замену подстроки без учета регистра.
- c. чтобы разделить строку на подстроки по определенному множеству разделителей.

Примечание: Разделитель - это последовательность из одного или нескольких символов, используемая для указания границы между отдельными независимыми областями в виде обычного текста или других потоков данных. Примером разделителя является символ запятой, который действует как разделитель полей в последовательности значений, разделенных запятыми.

15. Напишите программу на Python

- a. для вставки пробелов между словами, начинающимися с заглавных букв.
- b. для удаления из строки управляющих последовательностей ANSI.

16. Напишите программу на Python, чтобы удалить область символов в круглых скобках в строке.

Пример данных:

example (.com) w3resource «github (.com) «stackoverflow (.com)

Ожидаемый результат:

example w3resource github stackoverflow

17. Напишите программу на Python, чтобы соединить последовательные числа в данной строке.

Исходная строка:

Введите на 1 20 Kearny Street. Служба безопасности может направить вас на этаж 1 6.

Пожалуйста, подготовьте удостоверение личности.

После объединения последовательных чисел в указанной строке:

Вход по адресу 120 Kearny Street. Служба безопасности направит вас на 16 этаж.

Пожалуйста, подготовьте удостоверение личности.

18. Напишите программу на Python, которая проверяет, начинается ли слово и заканчивается ли оно гласным в данной строке. Возвращает true, если слово соответствует условию; в противном случае возвращает false.

Пример данных:

("Red Orange White") -> True

("Red White Black") -> False

("abcd dkise eosksu") -> True

19. Напишите программу на Python, которая берет строку с некоторыми словами. Для двух последовательных слов в указанной строке проверьте, заканчивается ли первое слово гласным, а следующее слово начинается с гласного. Если программа удовлетворяет условию, вернуть true, в противном случае false. Между словами допускается только один пробел.

Пример данных:

("These exercises can be used for practice.") -> True

("Following exercises should be removed for practice.") -> False