

9.3 Методы строк. Часть 1

Шаг 1

Тема урока: строки

1. Метод `capitalize()`
2. Метод `swapcase()`
3. Метод `title()`
4. Метод `lower()`
5. Метод `upper()`
6. Решение задач

Аннотация. Строковый тип данных, основные методы конвертации регистра.

Методы и функции

Мы уже знакомы с некоторыми встроенными функциями: `min()`, `max()`, `len()`, `int()`, `float()` и так далее. Метод – специализированная функция, тесно связанная с объектом. Как и функция, метод вызывается для выполнения отдельной задачи, но он вызывается для определенного объекта и “знает” о своем целевом объекте во время выполнения.

Таким образом: метод – функция, применяемая к объекту. В данном случае к строке. Метод вызывается в виде `имя_объекта.имя_метода(параметры)`.

Например, `s.find('e')` – это применение к строке `s` метода `find` с одним параметром `'e'`.

Методы строкового типа данных можно разделить на три группы:

1. Конвертация регистра;
2. Поиск и замена;
3. Классификация символов.

Конвертация регистра

Методы в этой группе выполняют преобразование регистра для строк.

Метод `capitalize()`

Метод `capitalize()` возвращает копию строки `s`, в которой первый символ имеет верхний регистр, а все остальные символы имеют нижний регистр.

Приведённый ниже код:

```
s = 'fo0 BaR BAZ quX'  
print(s.capitalize())
```

выводит:

```
Foo bar baz qux
```

Символы, не являющиеся буквами алфавита, остаются неизменными.

Приведённый ниже код:

```
s = 'foo123#BAR#.'  
print(s.capitalize())
```

выводит:

```
Foo123#bar#.
```

Метод swapcase()

Метод `swapcase()` возвращает копию строки `s`, в которой все символы, имеющие верхний регистр, преобразуются в символы нижнего регистра и наоборот.

Приведённый ниже код:

```
s = 'FOO Bar 123 baz qUX'  
print(s.swapcase())
```

выводит:

```
foo bAR 123 BAZ Qux
```

Метод title()

Метод `title()` возвращает копию строки `s`, в которой первый символ каждого слова переводится в верхний регистр.

Приведённый ниже код:

```
s = 'the sun also rises'  
print(s.title())
```

выводит:

```
The Sun Also Rises
```

Этот метод использует довольно простой алгоритм: он не пытается различить важные и неважные слова и не обрабатывает аббревиатуры и апострофы.

Приведённый ниже код:

```
s = "what's happened to ted's IBM stock?"  
print(s.title())
```

выводит:

```
What'S Happened To Ted'S Ibm Stock?
```

Метод lower()

Метод `lower()` возвращает копию строки `s`, в которой все символы имеют нижний регистр.

Приведённый ниже код:

```
s = 'FOO Bar 123 baz qUX'  
print(s.lower())
```

выводит:

```
foo bar 123 baz qux
```

Метод upper()

Метод `upper()` возвращает копию строки `s`, в которой все символы имеют верхний регистр.

Приведённый ниже код:

```
s = 'FOO Bar 123 baz qUX'  
print(s.upper())
```

выводит:

```
F00 BAR 123 BAZ QUX
```



Одно очень важное замечание о методах данной категории состоит в том, что они не изменяют исходную строку.

Если вы хотите изменить строку `s`, нужно написать код: `s = s.lower()`. На самом деле тут вы создаёте совсем другой объект в памяти компьютера, просто он со старым названием `s`.

Примечание

Англо-русский словарик:

`capitalize` – писать прописными буквами, закрепить.

`swapcase` – обменять регистр. `swap` – гл. обмениваться, `case` – случай, регистр, падеж, дело, расследование...

`title` – заголовок, титул.

`lower` – нижний.

`upper` – верхний.

❤️ Happy Pythoning! 🐍

Шаг 2

Что покажет приведённый ниже код?

```
s = 'i Learn Python language'  
print(s.capitalize())
```

Чтобы решить это задание откройте <https://stepik.org/lesson/296416/step/2>

Шаг 3

Что покажет приведённый ниже код?

```
s = 'i LEARN Python LAnguaGE'  
print(s.lower())
```

Чтобы решить это задание откройте <https://stepik.org/lesson/296416/step/3>

Шаг 4

Что покажет приведённый ниже код?

```
s = '$12344%#$@!'
print(s.lower())
```

Чтобы решить это задание откройте <https://stepik.org/lesson/296416/step/4>

Шаг 5

Что покажет приведённый ниже код?

```
s1 = 'a'
s2 = s1.upper()
print(s1, s2)
```

Чтобы решить это задание откройте <https://stepik.org/lesson/296416/step/5>

Шаг 6

Что покажет приведённый ниже код?

```
s = 'i LEARN Python LAnguaGE'
print(s.upper())
```

Чтобы решить это задание откройте <https://stepik.org/lesson/296416/step/6>

Шаг 7

Что покажет приведённый ниже код?

```
s = 'i LEARN Python LAnguaGE'
print(s.swapcase())
```

Чтобы решить это задание откройте <https://stepik.org/lesson/296416/step/7>

Шаг 8

Заглавные буквы

На вход программе подаётся строка, состоящая из имени и фамилии человека, разделённых одним пробелом. Напишите программу, которая проверяет, что имя и фамилия начинаются с заглавной буквы.

Формат входных данных

На вход программе подаётся строка.

Формат выходных данных

Программа должна вывести «YES» (без кавычек), если имя и фамилия начинаются с заглавной буквы, или «NO» (без кавычек) в противном случае.

Примечание. Стока содержит только буквы и символ пробела.

► Тестовые данные 

Sample Input 1:

chris alan

Sample Output 1:

NO

Sample Input 2:

Chris Alan

Sample Output 2:

YES

Sample Input 3:

chris Alan

Sample Output 3:

NO

Чтобы решить это задание откройте <https://stepik.org/lesson/296416/step/8>

Шаг 9

sWAP cASE

На вход программе подаётся строка. Напишите программу, которая меняет регистр символов – заменяет все строчные символы заглавными и наоборот.

Формат входных данных

На вход программе подаётся строка.

Формат выходных данных

Программа должна вывести строку в соответствии с условием задачи.

► Тестовые данные 

Sample Input 1:

HTTPS://pygen.RU/

Sample Output 1:

https://PYGEN.ru/

Sample Input 2:

Www.sTepik.com

Sample Output 2:

Sample Input 3:

Pythonist 2

Sample Output 3:

pYTHONIST 2

Чтобы решить это задание откройте <https://stepik.org/lesson/296416/step/9>

Шаг 10

Хороший оттенок

На вход программе подаётся строка текста. Напишите программу, которая определяет, является ли оттенок текста хорошим или нет. Текст имеет хороший оттенок, если содержит подстроку «хорош» (без кавычек) во всевозможных регистрах.

Формат входных данных

На вход программе подаётся строка текста.

Формат выходных данных

Программа должна вывести «YES» (без кавычек), если текст имеет хороший оттенок, или «NO» (без кавычек) в противном случае.

Примечание. Текст, содержащий «хорош», «ХОРОШ», «Хорош», «хОРОШ» и так далее также имеет хороший оттенок.

► Тестовые данные 

Sample Input 1:

я очень хороший текст =)

Sample Output 1:

YES

Sample Input 2:

оыралоывало ХОРОШвмсва выарлвօ3կգ834

Sample Output 2:

YES

Sample Input 3:

Цвет настроения синий

Sample Output 3:

NO

Чтобы решить это задание откройте <https://stepik.org/lesson/296416/step/10>

Нижний регистр

На вход программе подаётся строка. Напишите программу, которая подсчитывает количество буквенных символов в нижнем регистре.

Формат входных данных

На вход программе подаётся строка.

Формат выходных данных

Программа должна вывести количество буквенных символов в нижнем регистре.

► Тестовые данные 

Sample Input 1:

abcABCD12345

Sample Output 1:

3

Sample Input 2:

gggggggg1212321ABDCEFCE

Sample Output 2:

8

Sample Input 3:

2376423745dhdhdPPPP

Sample Output 3:

5

Чтобы решить это задание откройте <https://stepik.org/lesson/296416/step/11>