

Python: Формирование строк в циклах

Еще циклы можно использовать, чтобы формировать строки. Подобная задача часто встречается в веб-программировании. Она сводится к обычной агрегации, когда применяется интерполяция или конкатенация.

Переворот строки — алгоритмическая задача, которую задают на собеседованиях. Правильный способ перевернуть строку — использовать функцию из стандартной библиотеки. Но важно знать, как ее реализовать.

Один из алгоритмов выглядит так:

1. Строим новую строку
2. Перебираем символы исходной строки в обратном порядке

```
def reverse_string(string):  
    index = len(string) - 1  
    reversed_string = ''  
  
    while index >= 0:  
        current_char = string[index]  
        reversed_string = reversed_string + current_char  
        # То же самое через интерполяцию
```

```
# reversed_string = f'{reversed_string}{current_cha
index = index - 1

return reversed_string

reverse_string('Game Of Thrones') # 'senorhT fO emaG'
# Проверка нейтрального элемента
reverse_string('') # ''
```

Разберем функцию построчно:

Советуем скопировать эту функцию в <https://repl.it/languages/python3> и поэкспериментировать с ней.

Работая со строками, программисты часто допускают ошибку — выходят за границы строки. Если неправильно подобрать начальное значение счетчика или допустить ошибку в предикате цикла, функция может обращаться к несуществующему символу.

Особенно часто забывают, что индекс последнего элемента всегда меньше на единицу размера строки. В строках начальный индекс равен 0, значит, индекс последнего элемента — `len(str) - 1`.

Задание

Реализуйте функцию `my_substr()`, которая извлекает из строки подстроку указанной длины. Она принимает на вход два аргумента: строку и длину, и возвращает

подстроку, начиная с первого символа:

Пример вызова:

```
string = 'If I look back I am lost'
print(my_substr(string, 1)) # => 'I'
print(my_substr(string, 7)) # => 'If I lo'
```

Используйте тот же подход, что в функции для переворота строки из урока: собирайте строку-результат в цикле, перебирая начальную строку до определённого момента.

Эту задачу можно решить при помощи слайсов. Но в этом упражнении мы хотим потренироваться в использовании циклов и поэтому реализуем эту функциональность самостоятельно. Именно так внутри и работают слайсы.

► Упражнение не проходит проверку — что делать?



► В моей среде код работает, а здесь нет 🤔

► Мой код отличается от решения учителя 🤔

► Прочитал урок — ничего не понятно 🤔

Полезное

- [Итерация](#)