

11.7 Списочные выражения

Шаг 1

Тема урока: списочные выражения

1. Списочные выражения
2. Решение задач

Аннотация. Списочные выражения. Создание списков без явного использования циклов и вызова списочного метода `append()`.

Создание списков

Для того чтобы создать список, состоящий из 10 нулей, мы можем использовать приведённый ниже код:

```
zeros = []
for i in range(10):
    zeros.append(0)
```

В Python, однако, есть более простой и компактный способ для создания такого списка. Мы можем использовать оператор умножения списка на число:

```
zeros = [0] * 10
```

Для создания списков, заполненных по более сложным правилам, нам приходится явно использовать цикл `for`.

Например, для создания списка целых чисел от 0 до 9 мы вынуждены писать такой код:

```
numbers = []
for i in range(10):
    numbers.append(i)
```

Приведённый выше код хоть и не является сложным, однако достаточно громоздок.

Списочные выражения

В Python есть механизм для создания списков из неповторяющихся элементов. Такой механизм называется **списочное выражение** (list comprehension).

Предыдущий код можно записать следующим образом:

```
numbers = [i for i in range(10)]
```

Общий вид списочного выражения следующий:

```
[выражение for переменная in последовательность]
```

где `переменная` – имя некоторой переменной, `последовательность` – последовательность значений, которые она принимает (список, строка или объект, полученный при помощи функции `range`), `выражение` – некоторое выражение, как правило, зависящее от использованной в списочном выражении переменной, которым будут заполнены элементы списка.

Примеры использования списочных выражений

1. Создать список, заполненный 10 нулями можно и при помощи списочного выражения:

```
zeros = [0 for i in range(10)]
```

2. Создать список, заполненный квадратами целых чисел от 0 до 9 можно так:

```
squares = [i ** 2 for i in range(10)]
```

3. Создать список, заполненный кубами целых чисел от 10 до 20 можно так:

```
cubes = [i ** 3 for i in range(10, 21)]
```

4. Создать список, заполненный символами строки:

```
chars = [c for c in 'abcdefg']  
print(chars)
```

Считывание входных данных

При решении многих задач из предыдущих уроков мы считывали начальные данные (строки, числа) и заполняли ими список. С помощью списочных выражений процесс заполнения списка можно заметно сократить.

Например, если сначала вводится число `n` – количество строк, а затем на вход поступают сами строки, то создать список можно так:

```
n = int(input())  
lines = [input() for _ in range(n)]
```

Можно опустить описание переменной `n`:

```
lines = [input() for _ in range(int(input()))]
```

Если требуется считать список чисел, то необходимо добавить преобразование типов:

```
numbers = [int(input()) for _ in range(int(input()))]
```

Обратите внимание, мы используем символ `_` в качестве имени переменной цикла, поскольку она не используется.



Списочные выражения часто используются для инициализации списков. В Python не принято создавать пустые списки, а затем заполнять их значениями, если можно этого избежать.

Условия в списочном выражении

В списочных выражениях можно использовать условный оператор. Например, если требуется создать список чётных чисел от 0 до 20, то мы можем написать такой код:

```
evens = [i for i in range(21) if i % 2 == 0]
```



Важно: для того, чтобы получить список, состоящий из четных чисел, лучше использовать функцию `range(0, 21, 2)`. Предыдущий пример приведен для демонстрации возможности использования условий в списочных выражениях.

Вложенные циклы

В списочном выражении можно использовать вложенные циклы.

Приведённый ниже код:

```
numbers = [i * j for i in range(1, 5) for j in range(2)]  
print(numbers)
```

ВЫВОДИТ:

```
[0, 1, 0, 2, 0, 3, 0, 4]
```

Приведённый выше код код равнозначен следующему:

```
numbers = []

for i in range(1, 5):
    for j in range(2):
        numbers.append(i * j)
print(numbers)
```

Подводя итог

Пусть word = 'Hello', numbers = [1, 14, 5, 9, 12], words = ['one', 'two', 'three', 'four', 'five', 'six'] .

Списочное выражение	Результирующий список
[0 for i in range(10)]	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
[i ** 2 for i in range(1, 8)]	[1, 4, 9, 16, 25, 36, 49]
[i * 10 for i in numbers]	[10, 140, 50, 90, 120]
[c * 2 for c in word]	['HH', 'ee', 'll', 'll', 'oo']
[m[0] for m in words]	['o', 't', 't', 'f', 'f', 's']
[i for i in numbers if i < 10]	[1, 5, 9]
[m[0] for m in words if len(m) == 3]	['o', 't', 's']

❤ Happy Pythoning! 🐍

Шаг 2

Как записывается списочное выражение?

Чтобы решить это задание откройте <https://stepik.org/lesson/326725/step/2>

Шаг 3

Используя списочное выражение, дополните приведённый ниже код так, чтобы получить новый список, содержащий строки исходного списка, где у каждой строки удалён первый символ.

Чтобы решить это задание откройте <https://stepik.org/lesson/326725/step/3>

Шаг 4

Используя списочное выражение, дополните приведённый ниже код так, чтобы получить новый список, содержащий длины строк исходного списка.

Чтобы решить это задание откройте <https://stepik.org/lesson/326725/step/4>

Шаг 5

Используя списочное выражение, дополните приведённый ниже код так, чтобы получить новый список, содержащий только слова длиной не менее пяти символов (включительно).

Чтобы решить это задание откройте <https://stepik.org/lesson/326725/step/5>

Шаг 6

Используя списочное выражение, дополните приведённый ниже код так, чтобы получить список всех чисел-палиндромов от \$100\$ до \$1000\$ (включительно).

Примечание. Результирующий список должен состоять из целых чисел.

Чтобы решить это задание откройте <https://stepik.org/lesson/326725/step/6>

Шаг 7

Списочное выражение 1

На вход программе подаётся натуральное число \$n\$. Напишите программу, использующую **списочное выражение**, которая создаёт список, содержащий квадраты чисел от \$1\$ до \$n\$ (включительно), а затем выводит его элементы построчно, то есть каждый на отдельной строке.


Формат входных данных

На вход программе подаётся натуральное число.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

Примечание. Для вывода элементов списка используйте цикл `for`.

► Тестовые данные 

Sample Input:

5

Sample Output:

1
4
9
16
25

Чтобы решить это задание откройте <https://stepik.org/lesson/326725/step/7>

Шаг 8

Списочное выражение 2

На вход программе подаётся строка текста, содержащая целые числа. Напишите программу, **использующую списочное выражение**, которая выведет кубы указанных чисел на одной строке.

Формат входных данных

На вход программе подаётся строка текста, содержащая целые числа, разделённые символом пробела.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

Примечание 1. Для вывода элементов списка используйте цикл `for`.

Примечание 2. Используйте метод `split()`.

► Тестовые данные ●

Sample Input 1:

2 4 3

Sample Output 1:

8 64 27

Sample Input 2:

-2 -5 0

Sample Output 2:

-8 -125 0

Чтобы решить это задание откройте <https://stepik.org/lesson/326725/step/8>

Шаг 9

В одну строку 1

На вход программе подаётся строка текста, содержащая слова. Напишите программу, которая выводит слова введённой строки в столбик.


Формат входных данных

На вход программе подаётся строка текста, содержащая слова, разделённые символом пробела.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

Примечание. Программу можно написать в одну строку кода.

► Тестовые данные 

Sample Input:

```
Умей ценить того кто без тебя не может
```

Sample Output:

```
Умей
ценить
того
кто
без
тебя
не
может
```

Чтобы решить это задание откройте <https://stepik.org/lesson/326725/step/9>

Шаг 10

В одну строку 2

На вход программе подаётся строка текста. Напишите программу, **использующую списочное выражение**, которая выводит все цифровые символы данной строки.


Формат входных данных

На вход программе подаётся строка текста.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

Примечание. Программу можно написать в одну строку кода.

► Тестовые данные 

Sample Input 1:

```
Число Pi примерно равно 3.1415
```

Sample Output 1:

```
31415
```

Sample Input 2:

```
123Python awesome!56
```

Sample Output 2:

```
12356
```

Чтобы решить это задание откройте <https://stepik.org/lesson/326725/step/10>

В одну строку 3

На вход программе подаётся строка текста, содержащая целые числа. Напишите программу, использующую списочное выражение, которая выведет квадраты чётных чисел, кроме тех квадратов, которые оканчиваются на цифру 4.


Формат входных данных

На вход программе подаётся строка текста, содержащая целые числа, разделённые символом пробела.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

Примечание. Программу можно написать в одну строку кода.

► Тестовые данные 

Sample Input 1:

```
1 2 3 4 5 6 7 8 9
```

Sample Output 1:

```
16 36
```

Sample Input 2:

```
4 4 10 6 4
```

Sample Output 2:

```
16 16 100 36 16
```

Чтобы решить это задание откройте <https://stepik.org/lesson/326725/step/11>