Урок 7.  
Цикл for и range.

# Функция range

range(старт, стоп, шаг)

# Возвращает список целых чисел в форме

[старт, старт + шаг, старт + шаг\*2 ..]

Функция range() используется для генерации ряда целых чисел в заданном диапазоне. В общем виде функция выглядит так: range(from, to, step), где from – это первое значение диапазона (целое число), to – граница диапазона (это число не включается!), а step – интервал между числами диапазона. В такой конфигурации получаем, что: from < to, step > 0.

Возможна и установка убывающего ряда. Тогда получаем: from > to, step < 0.

print(tuple(range(1, 20)))

print(tuple(range(-100, -120, -1)))

Несколько деталей:

1. Функция range возвращает объект range, поэтому, чтобы вывести данные в том виде, в каком мы желаем, следует преобразовать этот объект в кортеж чисел.
2. Если значение from равно нулю, его можно не указывать и просто писать, например, range(10).

Пользуясь базовыми примерами, решите задачи ниже:

**Задача “Вывести диапазон”**

**Задача “Вывести четные”**

# 

# Цикл for

Циклы – наиболее часто употребляемые конструкции в любом языке. Они позволяют анализировать информацию, перебирать значения, сортировать и т. п. На сегодняшнем занятии мы поговорим о цикле for. Этот цикл удобен, когда мы точно знаем, сколько итераций (повторений определенного блока команд) следует сделать.

for <элемент> in <последовательность>:

<выражение>

**Переменной** присваивается значение первого элемента [последовательности](https://smartiqa.ru/courses/python/lesson-2" \l "sequence), после чего выполняется **выражение**(или несколько выражений) внутри цикла. Затем **переменной** присваивается следующее по порядку значение и так далее до тех пор, пока не будут перебраны все элементы **последовательности**.   
Рассмотрим, как он используется на примере:

|  |
| --- |
| **for** i **in** range(10):  print(i) |

Данный фрагмент просто выводит все числа от нуля до девяти. Можно заметить несколько особенностей:

1. Присутствует двоеточие, и блок выполняемого кода помещается в отступы, аналогично с условными конструкциями.
2. Вводится переменная i. На каждой итерации мы перебираем одно из значений диапазона range(10), и это значение заносится в переменную. Таким образом, мы можем обращаться к i внутри блока, относящегося к циклу: например, в функции print().

# Примеры использования

Приведем несколько типичных примеров использования цикла for, чтобы Вы могли привыкнуть к нему перед решением задач:

1. Вывод нечетных чисел из диапазона, который вводится с клавиатуры (a < b):

|  |
| --- |
| a = int(input()) b = int(input()) **for** i **in** range(a, b):  **if** i % 2 != 0:  print(i) |

1. Вывод чисел, оканчивающихся на 5, из диапазона в обратном порядке:

|  |
| --- |
| a = int(input()) b = int(input()) **for** i **in** range(b, a, -1):  **if** i % 10 == 5:  print(i) |

Например:

Python

# Перебираем буквы в строке

word = 'Слово'for letter in word:

print(letter)

# Результат выполнения

С

л

о

в

о

**Операторы continue и break**

Оператор **continue** начинает следующий проход цикла, минуя оставшееся тело цикла ( **for** или **while**). Пример:   
**for letter in 'Стол':   
\_\_\_\_if letter == 'т':   
\_\_\_\_\_\_\_\_continue   
\_\_\_\_print(letter)**   
  
Оператор **break** досрочно прерывает цикл. Пример:   
**for letter in 'Стол':   
\_\_\_\_if letter == 'т':   
\_\_\_\_\_\_\_\_break   
\_\_\_\_print(letter)**

## Напишите программу на Python, которая печатает все числа от 0 до 6, кроме 3 и 6.

## Среднее арифметическое

* **Последовательно вводятся ненулевые числа. Определить сумму положительных и сумму отрицательных чисел. Закончить ввод чисел при вводе 0.**
* **Вывести на экран циклом пять строк из нулей, причем каждая строка должна быть пронумерована**

## Посчитать сумму числового ряда от 0 до 14 включительно. Например, 0+1+2+3+…+14

## Цвета радуги

﻿   Пятилетний Петя учит цвета радуги. Напишите программу, которая принимает на вход натуральное число n и печатает первые n цветов радуги. При этом, если n > 7, программа должна ответить "Радуга состоит только из семи цветов".

## Таблица умножения

Дано натуральное число *n*. Напишите программу, которая выводит таблицу умножения на *n*.

**Формат входных данных**  
На вход программе подается натуральное число.

**Формат выходных данных**  
Программа должна вывести таблицу умножения на введённое число.

**Sample Input 1:**

5

**Sample Output 1:**

5 x 1 = 5

5 x 2 = 10

5 x 3 = 15

5 x 4 = 20

5 x 5 = 25

5 x 6 = 30

5 x 7 = 35

5 x 8 = 40

5 x 9 = 45

5 x 10 = 50