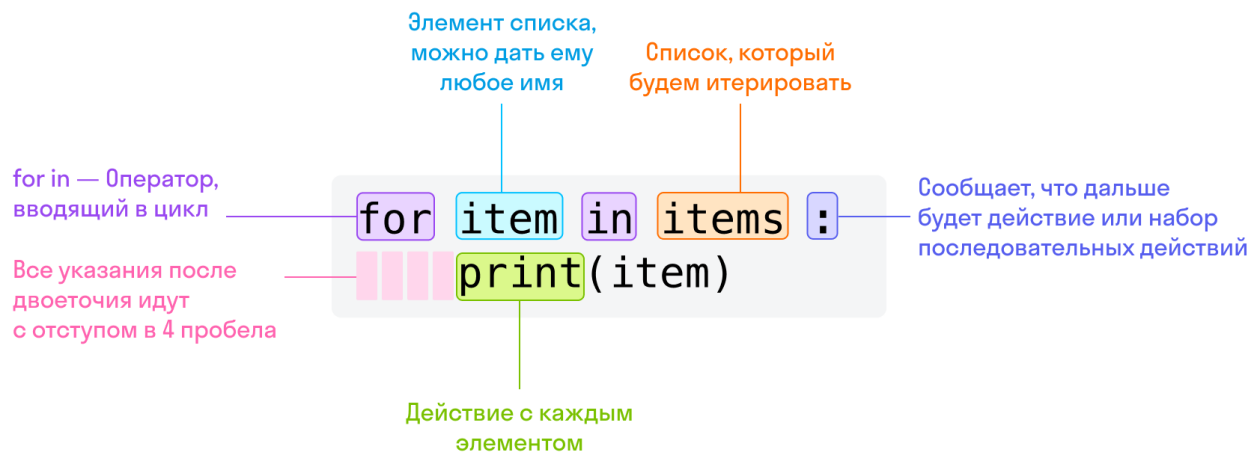


Циклы

Синтаксис цикла `for ... in`

```
for element in sequence:  
    # тело цикла
```



Как перебрать элементы списка

```
items_list = ["f", "f", "g", "h"]  
  
for item in items_list:  
    print(item)  
  
>>> f  
      f  
      g  
      h
```

Как вывести элементы списка, соответствующие условию

```
items_list = ["f", "f", "g", "h"]

for item in items_list:
    if item == "f":
        print(item)

>>> f
      f
```

```
items_list = ["f", "f", "g", "h"]

for item in items_list:
    if item != "f":
        print(item)

>>> g
      h
```

```
numbers = [1, 2, 3, 4, 5]
for number in numbers:
    if number > 3:
        result = number * 2
        print(result)

>>> 8
      10
```

Как посчитать сумму элементов списка

```
numbers = [1, 2, 3, 4, 5]
total = 0

for num in numbers:
    total += num

print(total)

>>> 15
```

Как посчитать количество итераций в цикле

```
numbers = [10, 20, 30, 40, 50]
count = 0

for num in numbers:
    count += 1

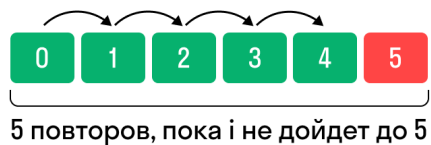
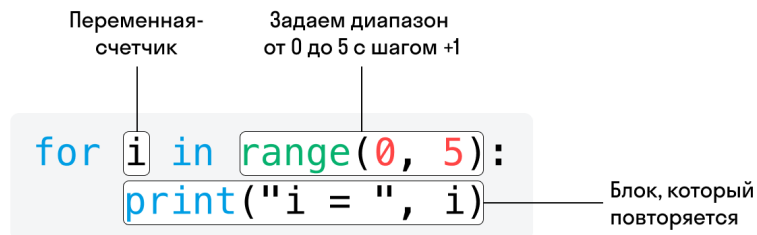
print(count)

>>> 5
```

Синтаксис функции `range()`

```
range(start, stop, step)
```

Конструкция `for in range()`



Создаем последовательность чисел с помощью `range()`

```
for num in range(4):  
    print(num)
```

```
>>> 0  
     1  
     2  
     3
```

Синтаксис цикла для перебора индексов списка

```
for index in range(len(list))
```

Как перебрать все элементы списка по индексам

```
fruits = ['яблоко', 'банан', 'груша']
```

```
for index in range(len(fruits)):  
    current_fruit = fruits[index]  
    print(current_fruit)
```

```
>>> 'яблоко'  
     'банан'  
     'груша'
```

Как пронумеровать элементы списка с единицы

```
girls = ["Иветта", "Лизетта", "Мюзетта"]
```

```
for i in range(len(girls)):  
    print(i+1, girls[i])
```

```
>>> 1 Иветта  
     2 Лизетта  
     3 Мюзетта
```

Синтаксис цикла `while`

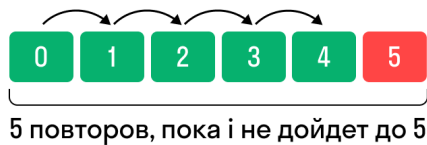
```
while условие:  
    # Тело цикла  
    # Этот блок кода выполняется, пока условие верно
```

Как посчитать до определенного числа с помощью цикла `while`

```
while i < 5:  
    print("i = ", i)  
    i += 1
```

Переменная-счетчик

Делаем инкремент с шагом 1



Как запустить цикл `while`, пока пользователь не введет нужные данные

```
user_input = input()  
while user_input != "exit":  
    user_input = input("Введите 'exit', чтобы выйти: ")
```

Как запустить бесконечный цикл

```
while True:  
    # Тело бесконечного цикла  
    # Будет выполняться бесконечно, пока не будет завершен (прерван)  
    # т. е. пока не выполнится условие остановки и не будет вызван оператор break
```

Как запустить бесконечный цикл с прерыванием оператором `break`

```
while True:  
    user_input = input("Введите 'exit', чтобы выйти: ")  
    if user_input == 'exit':  
        break  
    print("Вы ввели:", user_input)
```

Как защитить цикл от бесконечного выполнения оператором `break`

```
count = 0
while True:
    count += 1
    if count > 10:
        break
    print("Итерация номер", count)
```

Как остановить цикл при выполнении поиска элемента в списке

```
numbers = [1, 2, 3, 4, 5, 6]
target = 4

for num in numbers:
    if num == target:
        print("Элемент найден!")
        break
```

Как пропустить итерацию при определенном условии в цикле с оператором `continue`

```
count = 0
while count < 5:
    count += 1
    if count % 2 == 1:
        continue
    print("Это четное число:", count)
```

Как пронумеровать элементы списка с помощью `enumerate`

```
fruits = ["яблоко", "банан", "груша", "клубника"]

for index, fruit in enumerate(fruits):
    print(f"Индекс {index}: {fruit}")
```

Как найти элемент по индексу с помощью `enumerate`

```
fruits = ["яблоко", "банан", "груша", "клубника"]
target = "груша"

for index, fruit in enumerate(fruits):
    if fruit == target:
        print(f"Элемент '{target}' найден по индексу {index}")
        break
```

В чём отличие цикла `for` от `while`

for	while
Знаем количество элементов	Не знаем количество элементов
Знаем количество повторений	Не знаем количество повторений
	Знаем условие остановки

Как выбрать цикл

