



На этом уроке

- 1. Именные функции.
- 2. Оператор return.
- 3. Аргументы функций.
- 4. Анонимные функции.
- 5. Еще раз о встроенных функциях.
- 6. Функция range для многократно выполняемых действий.
- 7. Области видимости переменных в функциях.
- 8. Документирование кода функций.
- 9. Алгоритм создания функции.

Именные функции

<u>Функция определяется с помощью инструкции def</u>,

после которой следует имя функции.

```
def my_sum(arg_1, arg_2):
res = arg_1 + arg_2
```

Оператор return

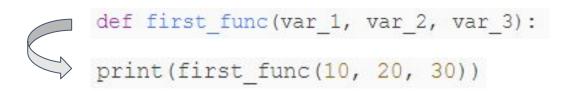
<u>Определяет выход из функции и передачу</u> <u>результата в точку вызова функции</u>

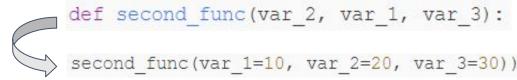
```
def my_sum(arg_1, arg_2):
    return arg_1 + arg_2
```

Аргументы функций

Позиционные

Именованные





Обязательные

Необязательные

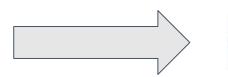
```
def first_func(var_1, var_2, var_3):
    print(first_func(10, 20, 30))
```

```
def second_func(var_1, var_2=20, var_3=30):
print(second_func(10))
```

Анонимные функции

Содержат только одно выражение,

выполняются быстрее именных



Еще раз о встроенных функциях

Операции с символами

Функция	Описание
ord()	Принимает Unicode-символ и возвращает соответствующий код (целое число)
chr()	Принимает целое число и возвращает Unicode- символ, соответствующий переданному числу (коду)
len()	Принимает любой объект-последовательность (строка, набор байтов, список, кортеж) или объект-коллекцию (словарь, множество) и возвращает число элементов последовательности

Еще раз о встроенных функциях

Математические функции

Функция	Описание
abs()	Принимает целое число или число с плавающей точкой. Возвращает абсолютное значение числа (по модулю)
round()	Принимает число с плавающей точкой. Округляет число до ближайшего целого числа. Также может принимать число знаков после запятой, до которых необходимо выполнить округление
divmod()	Принимает два числа, возвращает также два числа (частное и остаток от деления чисел)
pow()	Принимает два числа. Позволяет возвести первое число в указанную степень
max()	Принимает итерируемый объект и возвращает самый большой элемент
min()	Принимает итерируемый объект и возвращает наименьший элемент
sum()	Суммирует элементы последовательности

Функция range()

Отвечает за генерацию набора чисел в пределах

указанного диапазона.

```
print(list(range(7))) # целые числа в диапазоне [0, 7)
print(list(range(2, 8))) # целые числа в диапазоне [2, 8)
print(list(range(1, 9, 2))) # целые числа в диапазоне [1, 9) с шагом 2
print(list(range(1, -7, -2))) # целые числа в диапазоне [1, -7) с шагом -2
print(list(range(0))) # целые числа в диапазоне (0, 0)
print(list(range(1, 0))) # целые числа в диапазоне (1, 0)
```

Зоны видимости переменных

Локальная



Не локальная

Глобальная

Документирование кода функций

```
"""Возвращает путь до директории."""
Однострочное 
                       """Возвращает частное от деления.
                       Именованные параметры:
Многострочное 
                       р 1 -- делимое (по умолчанию 0.0)
                       р 2 -- делитель (по умолчанию 0.0)
```

Алгоритм создания функции

- 1. Придумать информативное имя функции.
- 2. Подготовить строки документации: назначение функции, типы данных параметров, тип данных результата.
- 3. Придумать информативные имена параметров функции.
- 4. Написать тело функции.



Итоги

Ha уроке вы научились ЭТОМ ваших представлять ЛОГИКУ программ в виде блоков-инструкций, которые можно, использовать функций. Это повторно, важнейший строительный элемент программ. Вы научились передавать в функцию параметры, выполнять ее вызов и возвращать результат. Вы познакомились анонимными функциями, а также таким понятием, как зона видимости. Наконец, вы некоторых полезных узнали встроенных функциях и об алгоритме создания собственных функций.