

Python

07

Функции

Функция это блок организованного, многократно используемого кода, который используется для выполнения конкретного задания. Функции обеспечивают лучшую модульность приложения и значительно повышают уровень повторного использования кода.

Функции

```
def my_func(a, b):  
    summ = a + b  
  
    print(f'{a} + {b} = {summ} ')  
  
    return summ  
  
my_func(4, 5)
```

Пример 7.01

Написать функцию, которая получает на вход имя и выводит строку вида: “Hello, {name}”.

Создать список из 5 имен. Вызвать функцию для каждого элемента списка в цикле.

Задание 7.02

Написать программу для работы с матрицами:

1. Создание
2. Вывод
3. Сумма всех элементов
4. Нахождение максимального элемента
5. Нахождение минимального элемента.

Задание 7.03

Написать программу для нахождения факториала. Факториал натурального числа n определяется как произведение всех натуральных чисел от 1 до n включительно

Аргументы по умолчанию

```
def my_format(string, char='!') :  
    result_string = f'{char}{string}{char}'  
    return result_string
```

Задание 7.04

Реализовать функцию возвращающую матрицу. На вход принимает n - размерность матрицы, `random_from`(по-умолчанию 1), `random_to`(по-умолчанию(9)).

Именованные аргументы

```
def my_pow(number, power) :  
    result = number ** power + 1  
    return result  
  
result = my_pow(power=3, number=5)  
print(result)
```

Сбор аргументов в коллекцию

```
def print_list(*args) :  
    print(args)  
  
print_list(1, 2, 3, 4)
```

Задание 7.05

Создать функцию, принимающая на вход неопределенное количество аргументом и возвращающая сумму $\text{args}[i] * i$

Пример: $\text{args} = [4, 3, 2, 1]$, $4 * 0 + 3 * 1 + 2 * 2 + 1 * 3 = 10$

Возврат функцией нескольких значений

```
def my_func(a, b):  
    summ = a + b  
    diff = a - b  
    return summ, diff
```

```
summ, diff = my_func(a, b)
```

Задание 7.06

Создать функцию, которая принимает на вход неопределенное количество аргументов и возвращает их сумму и максимальное из них.

Сбор аргументов в словарь

```
def my_func (**kwargs) :  
    for key, value in kwargs.items() :  
        print(key, value)  
  
my_func(a=5)
```

Задание 7.07

Создать функцию, которая принимает на вход неопределенное количество именных аргументов и выводит на экран те из них, длина ключа которых четна.

Обобщенное определение функции

```
def full_func(*args, **kwargs):  
    print(args)  
    print(kwargs)  
  
full_func(1, 2, 3, a=4, b=5, c=6)
```


Задание 7.08

Написать функцию принимающая на вход неопределенным количеством аргументов и именованный аргумент `mean_type`. В зависимости от `mean_type` вернуть среднеарифметическое либо среднегеометрическое. Написать программу в виде трех функций.