

**Модуль 3. Урок 1.**  
**ФУНКЦИИ**

алгоритмика

**Повторим**

# Сегодня на занятии:



- Функции — расширяем знания.
- Возвращаемые значения (заставим функцию всё вернуть).

# **Демонстрация**

**(программа с прошлого урока)**

алгоритмика

**Как реализовать  
двойную шифровку?**

# **Демонстрация**

**(объединение способов шифрования)**

**Что придётся сделать,  
если алгоритм нужно  
будет применить 3  
раза?**

# Демонстрация

(дублирование алгоритма)



алгоритмика

**Как решить  
проблему?**

# **Функция —**

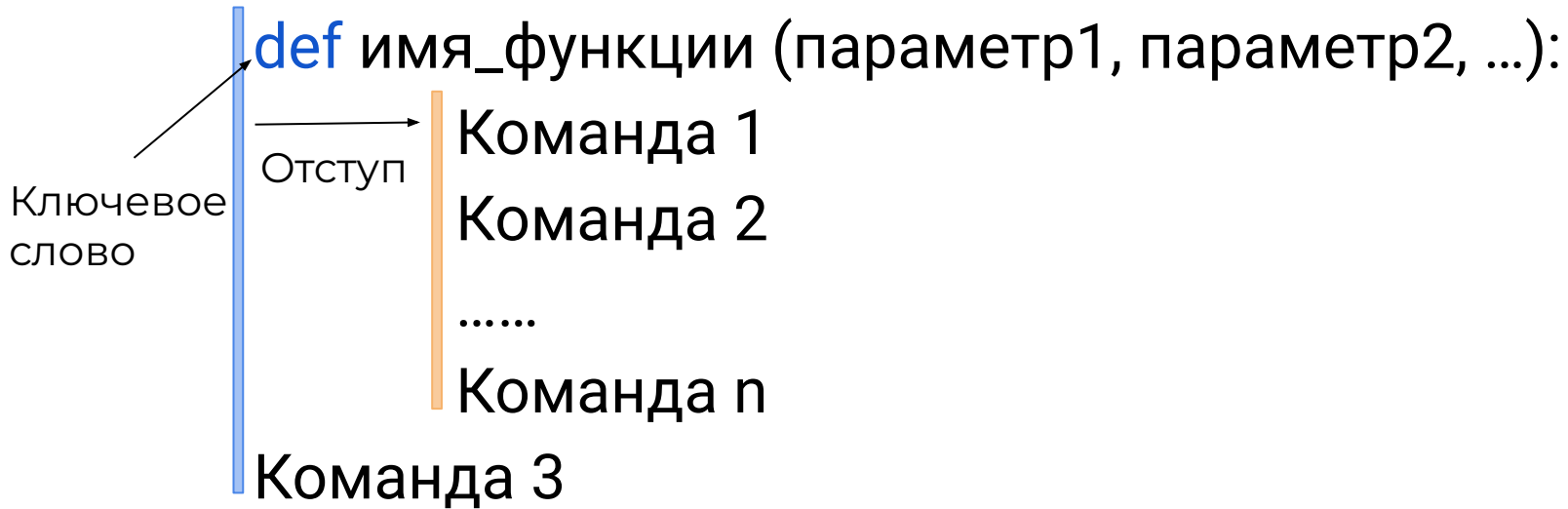
**это алгоритм, названный каким-либо именем.**

# Функции, которые мы уже использовали

`print("Привет!")` — выводит информацию на экран.

`input("Введите слово: ")` — возвращает данные от пользователя в программу.

# Создание функции



# Демонстрация

**(функция шифрования)**

# Демонстрация

**(функция с параметрами)**

# Параметры в созданной функции

```
def find (parametr1, parametr2):
```

```
    checklist3 = []
```

```
    for i in parametr1:
```

```
        for j in parametr2:
```

```
            if i == j:
```

```
                checklist3.append(i)
```

```
    print(checklist3)
```

```
checklist1 = input().split()
```

```
checklist2 = input().split()
```

```
find(checklist1, checklist2)
```

Обозначаем, что для работы функции нам потребуются некоторые данные из программы в процессе её выполнения, которые мы запишем в качестве двух параметров.

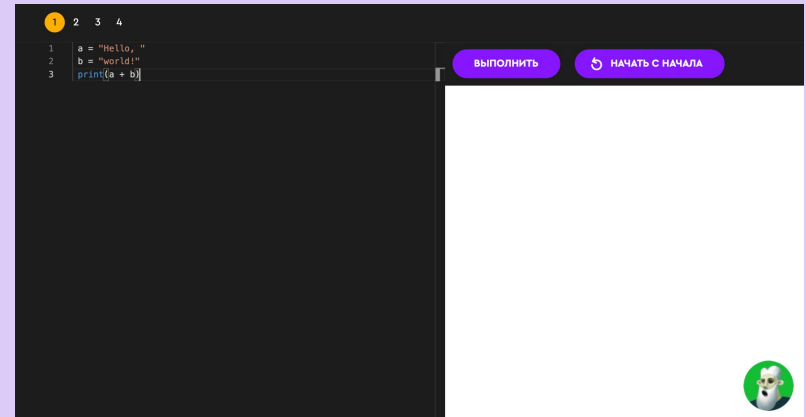
Присваиваем в качестве параметров переменные, которым присвоены данные, а конкретно — два списка.

# Заходим на платформу





# Создание функций



The image shows a code editor interface with a dark background. At the top, there are four numbered tabs (1, 2, 3, 4), with tab 1 being active and highlighted in yellow. The code editor contains three lines of Python code:

```
1 a = "Hello, "  
2 b = "world!"  
3 print(a + b)
```

To the right of the code editor, there are two purple buttons: "ВЫПОЛНИТЬ" (Execute) and "НАЧАТЬ С НАЧАЛА" (Start from beginning). Below the code editor, there is a large white area, likely for the output of the code. In the bottom right corner of the white area, there is a small circular icon of a white dog's head.

**Задание на платформе**

# Итог первой половины урока



алгоритмика

**Давайте отдохнём!**

# Как реализовать программу поиска наибольшего из двух чисел?

# **Демонстрация**

**(поиск наибольшего из двух чисел)**

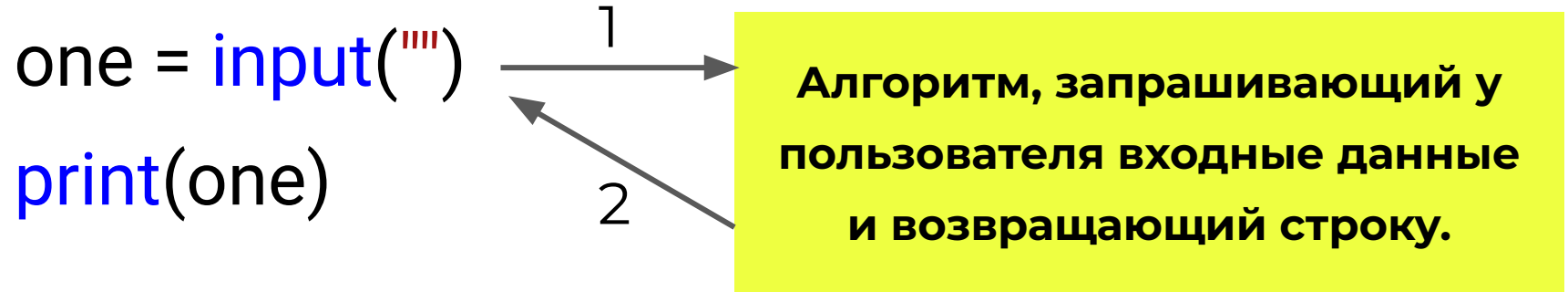
# Демонстрация

**(поиск наибольшего из двух чисел (функция))**

# Демонстрация

**(поиск наибольшего из двух чисел (сложение результатов с ошибкой))**

# Выполнение функции





# Демонстрация

**(input())**

**return —**

переводится как «вернуть». Команда для функции, позволяющая возвращать значение.

# Демонстрация

(пример с return)

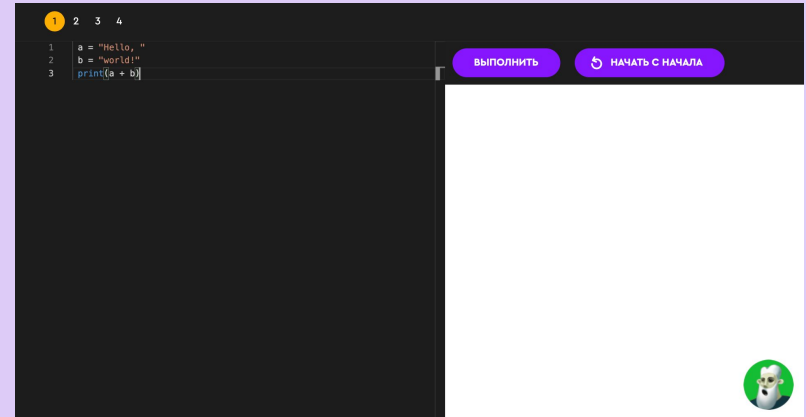
# Демонстрация

**(поиск наибольшего элемента (сложение результатов без ошибки))**

# Заходим на платформу



# Функции. Возвращаемые значения.



The image shows a code editor interface with a dark theme. At the top, there are four numbered tabs (1, 2, 3, 4), with tab 1 selected. The code in the editor is:

```
1 a = "Hello, "  
2 b = "world!"  
3 print(a + b]
```

Below the code, there are two buttons: "ВЫПОЛНИТЬ" (Execute) and "НАЧАТЬ С НАЧАЛА" (Run from Start). In the bottom right corner, there is a small circular icon of a white robot head.

**Задание на платформе**

алгоритмика

**Как прошло занятие?**

# Проверь себя

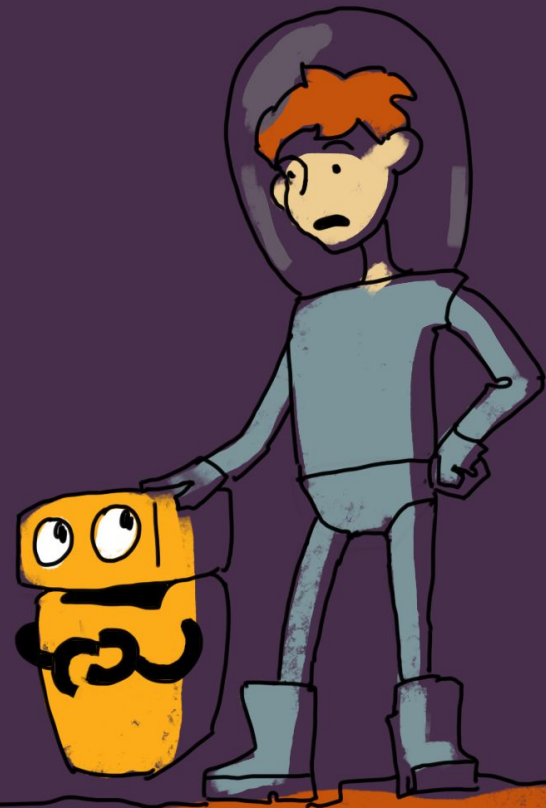
- Что такое функция?
- Что такое параметры функции?
- Как записать конструкцию для создания функции?
- Что такое возвращаемое значение?
- Какую команду использовать, чтобы функция возвращала значение?



# На следующем занятии:

- Модули и их создание — играем по крупному!

**До встречи!**



# Пример применения return:

```
def count(a):  
    return a * 2 + a  
print(count(5))
```

Вывод программы:

15