

Функция в Python

```
def my_function(input_var1, input_var2):  
    print(input_var1, input_var2)  
    return input_var1 + input_var2
```

Где:

`def` – ключевое слово для объявления функции

`my_function` – имя функции (бывают функции без имен)

`input_var1, input_var2` – входные параметры

`return` – ключевое слово, чтобы вернуть значение туда, где функция была вызвана

План занятия

- Ревизия
- git
- Функции
- Практика

VCS

git

Визуализация

<https://marklodato.github.io/visual-git-guide/index-en.html>

Интерактивный туториал

<https://try.github.io>

«Зачем нам нужны функции?»

Практика

Файл "functions.py"

Еще больше концепций
про функции

Функции могут принимать

- Нулевое количество аргументов
- Позиционные (обычные) аргументы
- Аргументы с параметром по-умолчанию (keyword)
- Любое количество позиционных аргументов
- Любое количество keyword-аргументов

Практика

Файл "functions/args_and_kwargs.py"

Полезные концепции из функционального программирования

- `map` - выполняет функцию для каждого объекта из массива входных параметров
- `filter` - оставляет из массива только что удовлетворяет условию
- `reduce` - выполняет функцию для каждого объекта из массива входных параметров, оперируя остатком

Практика

Файл "functions/fr.py"

«Функция - тоже объект».

—Знающие люди

Практика

Файл "functions/functions_are_objects.py"

Замыкание - возможность
функции использовать
чужие переменные

Практика

Файл "functions/enclosing.py"