

8-11 классы

Программирование на Python

Презентация занятия

Задачи по созданию собственных def и def с параметрами

41 занятие







Python: Задачи по созданию собственных def и def с параметрами. Тернарная операция

```
res1 = 'Greater' if a > b else 'Lower'
```

```
a = 10
b = 20
res1 = 'Greater' if a > b else 'Lower'
res2 = 'Equal' if a == b else 'Not equal'
print(res1) -> Lower
print(res2) -> Not equal
```





Python: Задачи по созданию собственных def и def с параметрами. filter

Встроенная функция filter имеет сигнатурку filter(func, iterable). В отличии от функции map, filter может принимать только функции, возвращающие логические значения bool.

```
def func(elem):
    return elem > 0
numbers = [-1, 2, -3, 4, 0, -20, 10]
positive_numbers = list(filter(func,
numbers))
print(positive_numbers)
```





Python: Задачи по созданию собственных def и def с параметрами.

Перепишите код программы с использованием lambda, map,

<u>filter и тернарной операции в одну строку:</u>

```
def func(a, b):
    s = a+b
    if s%10 > 5:
        return 'a'
    elif s%10 <=5:
        return 'b',
x = [5,7,3,6,8,0,3,4,7,9,0,2]
y = [9,6,3,6,8,3,6,8,4,5,7,4]
res=[]
for i in range(len(x)):
    res.append(func(x[i], y[i]))
print(res)
```





Python: Задачи по созданию собственных def и def с параметрами. Самостоятельная работа.

Необходимо реализовать $print_all$, принимающую произвольное количество позиционных аргументов и два именных, необязательных аргумента sep, end. Функция должна возвращать строку «готовую для вывода»:

```
print_all(1,2,3,4, sep='a', end='!') -> '1a2a3a4!'
print_all(1,2,3,4, end='##') -> '1 2 3 4##'
print_all(1,2,3,4, 5) -> '1 2 3 4 5\n'
```





Python: Задачи по созданию собственных def и def с параметрами. Самостоятельная работа.

Необходимо реализовать map_list , принимающую функцию для выполнения func, произвольное количество позиционных аргументов, каждый из которых содержит контейнеры. Функция должна возвращать список, состоящий из списков, на каждый элемент которого была применена функция func:

```
a, b, c = [1,2,3,4], [23,-1], [None, [25]]
print(map_list(str, a, b, c))
Выведет:
[['1', '2', '3', '4'], ['23', '-1'], ['None', '[25]']]
```







Python: Задачи по созданию собственных def и def с параметрами. Самостоятельная работа.

Необходимо реализовать $filter_list$, принимающую функцию для выполнения func, произвольное количество позиционных аргументов, каждый из которых содержит контейнеры. Функция должна возвращать список, состоящий из списков, каждый элемент которого удовлетворяет условию из функции func:

```
a, b, c= [1,2,3,4], [23,-1], [None, [25]]
numbers = lambda a: type(a) in (int, float)
print(filter_list(numbers, a, b, c))
Выведет:
[[1, 2, 3, 4], [23, -1], []]
```