

1. ЦЕЛЬ РАБОТЫ

- Освоить синтаксис объявления и вызова функций
 - Научиться передавать параметры и возвращать значения
 - Понять область видимости переменных
 - Сформировать навык декомпозиции задач
-

2. ТЕОРЕТИЧЕСКАЯ СПРАВКА

2.1. Базовый синтаксис

```
python
def имя_функции(параметры):
    """Строка документации"""
    тело_функции
    return результат
```

2.2. Типы параметров

- **Позиционные:** `def func(a, b)`
 - **Именованные:** `def func(name="Иван")`
 - **Произвольное число args:** `def func(*args)`
 - **Произвольное число kwargs:** `def func(**kwargs)`
-

3. СТРУКТУРА ЛАБОРАТОРНОЙ РАБОТЫ

ЗАДАНИЕ 1. Простые функции (базовый уровень)

Цель: научиться создавать и вызывать простые функции

Решите задачи:

1. Напишите функцию `greet(name)`, которая выводит приветствие (используйте f-строку)

2. Создайте функцию `square(x)`, возвращающую квадрат числа. Примените созданную функцию ко всем элементам списка `my_list = [-45, 46, 2, 5, 9, 6, -9, 117, 19, 21, 31]`. Получите новый список, содержащий квадраты чисел из первоначального списка.
3. Реализуйте функцию `is_simple(n)`, проверяющую является ли число простым. С помощью данной функции определите количество простых списка из предыдущей задачи
4. Реализуйте функцию `get_divisors()` поиска делителей числа. Протестируйте функцию на различных числах: 90, 155, 10000000000.

ВАРИАНТЫ

Вариант 1: Домик с полной кастомизацией

```
def draw_house(x, y, width, height, roof_height,
               base_color, roof_color, door_color):
    """
    Нарисовать кастомный домик

    Parameters:
    x, y - координаты начальной точки
    width - ширина основания
    height - высота стен
    roof_height - высота крыши
    base_color - цвет стен
    roof_color - цвет крыши
    door_color - цвет двери
    """
```

Вариант 2. Дерево (фрактальное)

python

```
def draw_tree(x, y, trunk_length, angle, depth, color):
    """
    Нарисовать фрактальное дерево

    Parameters:
    trunk_length - длина ствола
    angle - угол между ветвями
    depth - уровень рекурсии
    """
```

Вариант 3. Машинка

```
def draw_car(x, y, width, height, body_color, wheel_color):
    """Нарисовать машинку"""
    # Кузов + 2 колеса
    # Возможность добавить окна, фары
```

Вариант 4. Ёлка с украшениями

python

```
def draw_christmas_tree(x, y, height, levels, has_decorations):  
    """Нарисовать ёлку"""  
    # levels - количество ярусов  
    # has_decorations - добавить шарик, гирлянды
```

Вариант 5: БАЗОВЫЙ СНЕГОВИК

```
def draw_snowman(x, y, size):  
    """  
    Нарисовать простого снеговика из трёх кругов  
  
    Parameters:  
    x, y - координаты центра нижнего шара  
    size - радиус нижнего шара  
    """
```

Вариант 6: Звезда

```
def draw_star(x, y, size, points, color):  
    """  
    Нарисовать звезду с заданным количеством лучей  
  
    Parameters:  
    x, y - координаты центра звезды  
    size - размер звезды  
    points - количество лучей (5, 6, 7, 8...)  
    color - цвет заливки  
    """
```