

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №14
дисциплины «Программирование на Python»
Вариант 9

Выполнил:
Дудкин Константин Александрович
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»
направление «Программное
обеспечение средств вычислительной
техники и автоматизированных
систем»,
очная форма обучения

(подпись)

Руководитель практики:
Кандидат технических наук, доцент
кафедры инфокоммуникаций, доцент
Воронкин Роман Александрович

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Тема: Замыкания в языке Python

Цель: Приобретение навыков по работе с замыканиями при написании программ с помощью языка программирования Python версии 3.x

Порядок выполнения работы

1. Выполнил индивидуальное задание: Используя замыкания функций, объявите внутреннюю функцию, которая принимает в качестве аргумента коллекцию (список или кортеж) и возвращает или минимальное значение, или максимальное, в зависимости от значения параметра `type` внешней функции. Если `type` равен «max», то возвращается максимальное значение, иначе – минимальное. По умолчанию `type` должно принимать значение «max». Вызовите внутреннюю функцию замыкания и отобразите на экране результат ее работы.

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  2 usages new *
5  def outer_function(type='max'):
6      new *
7      def inner_function(collection):
8          if type == 'max':
9              return max(collection)
10         else:
11             return min(collection)
12     return inner_function
13
14  if __name__ == "__main__":
15      input_data = input("Введите значения через пробел: ").split()
16      input_numbers = [int(x) for x in input_data]
17      max_func = outer_function()
18      print("Максимальное значение:", max(input_numbers))
19      min_func = outer_function(type='min')
20      print("Минимальное значение:", min(input_numbers))
```

Рисунок 1. Код программы задачи

```
/usr/bin/python3.11 /home/code_raider/git/Python_LW14/Python Programs/Individual.py
Введите значения через пробел: 2 4 67 1 4 76
Максимальное значение: 76
Минимальное значение: 1
Process finished with exit code 0
```

Рисунок 2. Результат выполнения программы

Ответы на вопросы

1. Замыкание в программировании — функция, в теле которой присутствуют ссылки на переменные, объявленные вне тела этой функции в окружающем коде и не являющиеся ее параметрами

2. Замыкания языка Python реализованы так, что вложенные функции имеют доступ к переменным внешней функции даже после работы внешней функции

3. Область видимости Local относится к переменным, созданным и используемым внутри функции. Эти переменные недоступны вне функции

4. Enclosing область видимости относится к локальным переменным внутри функции и ее вложенных функций

5. Global относится к глобальным переменным уровня модуля. Доступны из любой функции в данном модуле, но не из других модулей при импорте

6. Built-in относится к уровню интерпретатора Python, включая встроенные функции, исключения и прочие сущности доступные в любом модуле без импорта

7. Пример использования замыканий:

```
def mul(a)
    def helper(b)
        return a* b
    return helper
```

```
new_mul5 = mul(5)
```

```
result = new_mul5(2)
```

В данном примере переменная `a` сохраняет свое значение внутри возвращаемой функции `helper`, создавая замыкание

8. Пример использование замыкания для построения иерархических данных:

```
tpl = lambda a, b: (a, b)
```

```
a = tpl(1, 2)
```

```
b = tpl(3, a)
```

```
c = tpl(a, b)
```

В этом примере `tpl` порождает структуру данных `((1, 2), (3, (1, 2)))`, используя замыкания для сохранения значений

Вывод: В ходе работы были получены навыки по работе с замыканиями при написании программ с использованием языка программирования Python версии 3.x