

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Кафедра инфокоммуникаций

**Отчет по лабораторной работе №3**

**Работа с функциями в языке Python**

**По дисциплине «Технологии программирования и алгоритмизация»**

Выполнил студент группы ИВТ-б-о-20-1

Галяс Д. И. « » \_\_\_\_\_ 20\_\_ г.

Подпись студента \_\_\_\_\_

Работа защищена « » \_\_\_\_\_ 20\_\_ г.

Проверил Воронкин Р. А. \_\_\_\_\_

(подпись)

Ставрополь 2021

**Цель работы:** приобретение навыков по работе с функциями при написании программ с помощью языка программирования Python версии 3.x.

**Ход работы:**

**Ссылка на репозиторий:** <https://github.com/DIMITRY-GALYAS1/Rabota-10.git>

Создал новый репозиторий на github, после клонировал его и создал в папке репозитория новый проект PyCharm.

1. Проработал пример из лабораторной работы.

```
C:\Users\lizeq\anaconda3\envs\pythonProject3\python.exe "C:/Users/lizeq/PycharmProjects/pythonProject3/Primer 1.py"
>>> add
Фамилия и инициалы? Некифоров Сергей
Должность? Директор
Год поступления? 2005
>>> add
Фамилия и инициалы? Гончаров Кирилл
Должность? Заместитель
Год поступления? 2020
>>> list
+-----+-----+-----+-----+
| No |          Ф.И.О.          |      Должность      |      Год      |
+-----+-----+-----+-----+
|  1 | Гончаров Кирилл          | Заместитель          |      2020      |
+-----+-----+-----+-----+
|  2 | Некифоров Сергей        | Директор             |      2005      |
+-----+-----+-----+-----+
>>> select 5
+-----+-----+-----+-----+
| No |          Ф.И.О.          |      Должность      |      Год      |
+-----+-----+-----+-----+
|  1 | Некифоров Сергей        | Директор             |      2005      |
+-----+-----+-----+-----+
```

Рисунок 1. Пример 1

2. Решил общие задания.

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4
5 def test():
6     z = int(input('Введите число:'))
7     if z > 0:
8         positive(z)
9     else:
10        negative(z)
11
12
13 def positive(z):
14     print(f"{z} положительное")
15
16
17 def negative(z):
18     print(f"{z} отрицательное")
19
20
21 if __name__ == '__main__':
22     test()
```

Primer 1 x zadanie\_1 x

C:\Users\lizeq\anaconda3\envs\pythonProject3\python.exe C:/Users/lizeq/PycharmProjects/pythonProject3/zadanie\_1.py

Введите число: 5

5 положительное

Process finished with exit code 0

Рисунок 2. Задание 1

```

C:\Users\lizeq\anaconda3\envs\pythonProject3\python.exe C:/Users/lizeq/PycharmProjects/pythonProject3/zadanie_2.py
Введите высоту цилиндра: 8
Введите радиус цилиндра: 9
Какую площадь найти: площадь боковой поверхности или полную площадь цилиндра?
Для нахождения площади бокой поверхности введите - 1
Для нахождения площади полной поверхности цилиндра введите - 2
>>>1
Площадь боковой повехности: 452.16

```

Рисунок 3. Задание 2

```

#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import sys

def cylinder():
    h = int(input('Введите высоту цилиндра: '))
    rad = int(input('Введите радиус цилиндра: '))

def circle():
    print('Площадь полной поверхности цилиндра: ',
          2 * 3.14 * h * rad + 2 * 3.14 * rad ** 2)

z = input('Какую площадь найти: площадь боковой '
          'поверхности или полную площадь цилиндра?\n'
          'Для нахождения площади бокой поверхности введите - 1\n'
          'Для нахождения площади полной поверхности цилиндра введите - 2\n'
          '>>>')

if z == '2':
    circle()
elif z == '1':
    print('Площадь боковой повехности: ', 2 * 3.14 * h * rad)
else:
    print(f"Неизвестная команда {z}", file=sys.stderr)

if __name__ == '__main__':
    cylinder()

```

Рисунок 4. Код задания 2

```

C:\Users\lizeq\anaconda3\envs\pythonProject3\python.exe C:/Users/lizeq/PycharmProjects/pythonProject3/zadanie_3.py
Введите число: 2
Произведение = 2
Введите число: 5
Произведение = 10
Введите число: 6
Произведение = 60
Введите число: 0
Произведение = 0

Process finished with exit code 0

```

Рисунок 5. Задание 3

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

def main():
    q = 1
    while True:
        z = int(input("Введите число: "))
        q *= z
        if q == 0:
            print("Произведение = 0")
            break
        else:
            print(f"Произведение = {q}")

if __name__ == '__main__':
    main()
```

Рисунок 6. Код задания 3

```
C:\Users\lizeq\anaconda3\envs\pythonProject3\python.exe C:/Users/lizeq/PycharmProjects/pythonProject3/zadanie_4.py
Введите строку: 4
4
Process finished with exit code 0
```

Рисунок 7. Задание 4

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

def get_input():
    z = input('Введите строку: ')
    test_input(z)

def test_input(z):
    try:
        str_to_int(z)
    except ValueError:
        print('Нельзя преобразовать в число')

def str_to_int(z):
    i = int(z)
    print_int(i)

def print_int(i):
    print(i)

if __name__ == '__main__':
    get_input()
```

Рисунок 8. Код задания 4

3. Выполнил индивидуальное задание.

```
C:\Users\lizeq\anaconda3\envs\pythonProject3\python.exe C:/Users/lizeq/PycharmProjects/pythonProject3/individual_1.1.py
Список команд:

add - добавить студента;
list - вывести список студентов;
select - вывести список студентов, имеющих оценку 2;
exit - завершить работу с программой.
>>> add
Фамилия и инициалы? Некифоров Сергей
Номер группы? 1
Успеваемость: 2 3 4 4 4
>>> add
Фамилия и инициалы? Гончаров Кирилл
Номер группы? 2
Успеваемость: 4 3 5 5 5
>>> list
+-----+-----+-----+-----+
| № |          Ф.И.О.          | Группа | Успеваемость |
+-----+-----+-----+-----+
| 1 | Гончаров Кирилл          | 2      | 4 3 5 5 5 |
| 2 | Некифоров Сергей        | 1      | 2 3 4 4 4 |
+-----+-----+-----+-----+
>>> select
* Некифоров Сергей группа № 1
>>> exit

Process finished with exit code 0
```

Рисунок 9. Индивидуальное задание

### Контрольные вопросы:

1. Каково назначение функций в языке программирования Python?

Функции можно сравнить с небольшими программками, которые сами по себе, т.е. автономно, не исполняются, а встраиваются в обычную программу.

2. Каково назначение операторов def и return ?

def – создаёт функцию, return – возвращает параметр из функции

3. Каково назначение локальных и глобальных переменных при написании функций в Python?

К глобальной переменной можно обратиться из локальной области видимости. К локальной переменной нельзя обратиться из глобальной области видимости, потому что локальная переменная существует только в момент выполнения тела функции.

4. Как вернуть несколько значений из функции Python?

Просто перечислить их через запятую в return

5. Какие существуют способы передачи значений в функцию?

1)Любая функция может обратиться к глобальной переменной. 2)В функцию можно передать значение при вызове: function(значение)

6. Как задать значение аргументов функции по умолчанию?

При определении функции, в скобках указать переменные и их значения:  
`function(параметр=значение)`

7. Каково назначение lambda-выражений в языке Python?

Python поддерживает интересный синтаксис, позволяющий определять небольшие однострочные функции на лету. Позаимствованные из Lisp, так называемые lambda-функции могут быть использованы везде, где требуется функция.

8. Как осуществляется документирование кода согласно PEP257?

PEP 257 описывает соглашения, связанные со строками документации python, рассказывает о том, как нужно документировать python код.

Строки документации - строковые литералы, которые являются первым оператором в модуле, функции, классе или определении метода. Такая строка документации становится специальным атрибутом `__doc__` этого объекта.

Все модули должны, как правило, иметь строки документации, и все функции и классы, экспортируемые модулем также должны иметь строки документации. Публичные методы (в том числе `__init__`) также должны иметь строки документации. Пакет модулей может быть документирован в `__init__.py`.

9. В чем особенность однострочных и многострочных форм строк документации?

Однострочная строка документации не должна быть "подписью" параметров функции / метода (которые могут быть получены с помощью интроспекции).

Многострочные строки документации состоят из однострочной строки документации с последующей пустой строкой, а затем более подробным описанием. Первая строка может быть использована автоматическими средствами индексации, поэтому важно, чтобы она находилась на одной строке и была отделена от остальной документации пустой строкой. Первая строка может быть на той же строке, где и открывающие кавычки, или на

следующей строке. Вся документация должна иметь такой же отступ, как кавычки на первой строке

**Вывод:** в ходе выполнения лабораторной работы приобрел навыки по работы с функциями при написании программ с помощью языка программирования Python версии 3.x.