

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное  
образовательное учреждение высшего образования**

**«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**Отчет по лабораторной работе №2.8**

**Работа с функциями в языке Python.**

**по дисциплине «Технологии программирования и алгоритмизации»**

Выполнил студент группы ИВТ-б-о-20-1

Хашиев Х.М. « » \_\_\_\_\_ 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.

### Ход работы: Пример 1

<https://github.com/Mirror-Shard/L2.8>

1. Создал репозиторий на github с лицензией MIT, добавил .gitignore и выбрал язык Python.

2. Проработал пример из учебника, пример добавления нового сотрудника:

```
Фамилия и инициалы? Хашиев Х. М.  
Должность? Президент  
Год поступления? 2012  
>>> list
```

No	Ф.И.О.	Должность	Год
1	Хашиев Х. М.	Президент	2012

Рисунок 1 – Добавление нового сотрудника

3. Затем добавил ещё одного сотрудника, программа отсортировала список по Фамилии:

```
>>> add  
Фамилия и инициалы? Евкуров Ю. П.  
Должность? Уборщик  
Год поступления? 2000  
>>> list
```

No	Ф.И.О.	Должность	Год
1	Евкуров Ю. П.	Уборщик	2000
2	Хашиев Х. М.	Президент	2012

Рисунок 2 – Сортировка списка

4. Команда `help` выводит список доступных команд:

```
>>> help
Список команд:

add - добавить работника;
list - вывести список работников;
select <стаж> - запросить работников со стажем;
help - отобразить справку;
exit - завершить работу с программой.
```

Рисунок 3 – Вывод списка команд

### Задание 1

Решить следующую задачу: основная ветка программы, не считая заголовков функций, состоит из двух строки кода. Это вызов функции `test()` и инструкции `if __name__ == '__main__':`. В ней запрашивается на ввод целое число. Если оно положительное, то вызывается функция `positive()`, тело которой содержит команду вывода на экран слова "Положительное". Если число отрицательное, то вызывается функция `negative()`, ее тело содержит выражение вывода на экран слова "Отрицательное"

1. При вводе отрицательного числа программа пишет, что оно отрицательно:

```
Введите число: -9
Число отрицательное
```

Рисунок 4 – Ввод отрицательного числа

2. При вводе положительного числа – наоборот:

```
Введите число: 5
Число положительное

Process finished with exit code 0
```

Рисунок 5 – Код второго задания

## Задание 2

Решите следующую задачу: в основной ветке программы вызывается функция `cylinder()`, которая вычисляет площадь цилиндра. В теле `cylinder()` определена функция `circle()`, вычисляющая площадь круга по формуле  $S = \pi r^2$ . В теле `cylinder()` у пользователя спрашивается, хочет ли он получить только площадь боковой поверхности цилиндра, которая вычисляется по формуле  $S_{\text{бок}} = 2\pi r h$ , или полную площадь цилиндра. В последнем случае к площади боковой поверхности цилиндра должен добавляться удвоенный результат вычислений функции `circle()`.

1. Ввёл радиус и высоту и выбрал вычислить площадь всего цилиндра:

```
Введите радиус: 8
Введите высоту: 15
Вычислить площадь всего цилиндра или только боковой стороны?
1 - всего цилиндра, 2 - только боковой стороны
1
Площадь всего цилиндра равна:
1156.1060965210438
```

Рисунок 6 – Вычисление площади всего цилиндра

2. Затем ввёл те же самые радиус и высоту, но выбрал вычислить только площадь боковой стороны

```
Введите радиус: 8
Введите высоту: 15
Вычислить площадь всего цилиндра или только боковой стороны?
1 – всего цилиндра, 2 – только боковой стороны
2
Площадь боковой стороны цилиндра равна:
753.9822368615503
```

Рисунок 7 – Вычисление боковой стороны

### Задание 3

Решите следующую задачу: напишите функцию, которая считывает с клавиатуры числа и перемножает их до тех пор, пока не будет введен 0. Функция должна возвращать полученное произведение. Вызовите функцию и выведите на экран результат ее работы.

1. Организовал бесконечный запрос и перемножение чисел:

```
a = 5
b = 6
30.0
a = 2
b = 3
6.0
a =
```

Рисунок 8 – Ход работы программы

2. Однако, при вводе нуля программа заканчивает свою работу:

```
b = 0
Адиос!
```

Рисунок 9 – Завершение программы

### Задание 4

Решите следующую задачу: напишите программу, в которой определены следующие четыре функции:

1. Функция `get_input()` не имеет параметров, запрашивает ввод с клавиатуры и возвращает в основную программу полученную строку.

2. Функция `test_input()` имеет один параметр. В теле она проверяет, можно ли переданное ей значение преобразовать к целому числу. Если можно, возвращает логическое `True`.

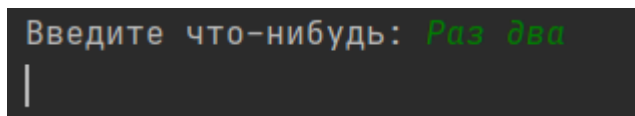
Если нельзя – `False`.

3. Функция `str_to_int()` имеет один параметр. В теле преобразовывает переданное значение к целочисленному типу. Возвращает полученное число.

4. Функция `print_int()` имеет один параметр. Она выводит переданное значение на экран и ничего не возвращает.

В основной ветке программы вызовите первую функцию. То, что она вернула, передайте во вторую функцию. Если вторая функция вернула `True`, то те же данные (из первой функции) передайте в третью функцию, а возвращенное третьей функцией значение – в четвертую.

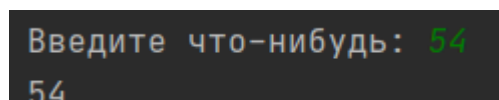
1. При вводе строки, программа останавливает работу на второй функции и ничего не делает:



```
Введите что-нибудь: Раз два
|
```

Рисунок 10 – Код задания 4

2. При вводе числа, программа преобразовывает его в `int` и выводит обратно:



```
Введите что-нибудь: 54
54
```

Рисунок 11 – Результат работы задания 4

### Индивидуальное задание

Решить индивидуальное задание лабораторной работы 2.6, оформив каждую команду в виде отдельной функции.

1. Переделал все команды из задания в отдельные функции, работа программы идёт так же корректно:

```
Фамилия и инициалы? Хашиев Х. М.  
Должность? Президент  
Год поступления? 2012  
>>> list
```

No	Ф.И.О.	Должность	Год
1	Хашиев Х. М.	Президент	2012

Рисунок 12 – Добавление нового сотрудника

2. Добавил ещё одного сотрудника:

```
>>> add  
Фамилия и инициалы? Евкуров Ю. П.  
Должность? Уборщик  
Год поступления? 2000  
>>> list
```

No	Ф.И.О.	Должность	Год
1	Евкуров Ю. П.	Уборщик	2000
2	Хашиев Х. М.	Президент	2012

Рисунок 13 – Сортировка списка

3. Команда help всё также выводит список доступных команд:

```
>>> help
Список команд:

add - добавить работника;
list - вывести список работников;
select <стаж> - запросить работников со стажем;
help - отобразить справку;
exit - завершить работу с программой.
```

Рисунок 14 – Вывод списка команд

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