

**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  
Основы языка Python 3  
по дисциплине «Основы кроссплатформенного программирования»**

Выполнил студент группы ИВТ-б-о-20-1

Токарев В. А. « » \_\_\_\_\_ 20\_\_ г.

Подпись студента \_\_\_\_\_

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

Проверил Воронкин Р. А. \_\_\_\_\_

(подпись)

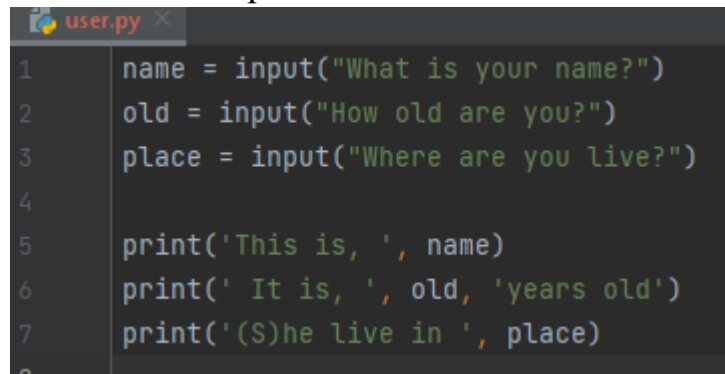
Ставрополь 2021

**Цель работы:** исследование процесса установки и базовых возможностей языка Python версии 3.

Ссылка на репозиторий: <https://github.com/UnDeR-The-mAsK/lab3.git>

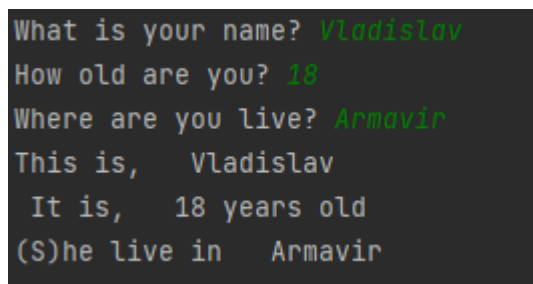
### Ход работы:

1. Создал новый репозиторий
2. Приступил к выполнению первого задания.



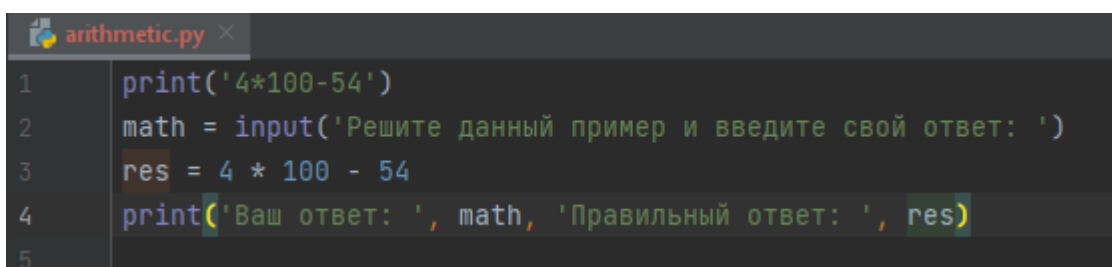
```
1 name = input("What is your name?")
2 old = input("How old are you?")
3 place = input("Where are you live?")
4
5 print('This is, ', name)
6 print(' It is, ', old, 'years old')
7 print('(S)he live in ', place)
8
```

Рис. 1 Информация обо мне



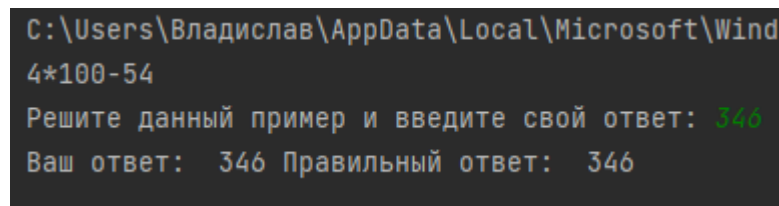
```
What is your name? Vladislav
How old are you? 18
Where are you live? Armavir
This is, Vladislav
It is, 18 years old
(S)he live in Armavir
```

Рис. 2 Результат выполнения кода



```
1 print('4*100-54')
2 math = input('Решите данный пример и введите свой ответ: ')
3 res = 4 * 100 - 54
4 print('Ваш ответ: ', math, 'Правильный ответ: ', res)
5
```

Рис. 3 Решение математического примера



```
C:\Users\Владислав\AppData\Local\Microsoft\Wind
4*100-54
Решите данный пример и введите свой ответ: 346
Ваш ответ: 346 Правильный ответ: 346
```

Рис. 4 Результат решения

```

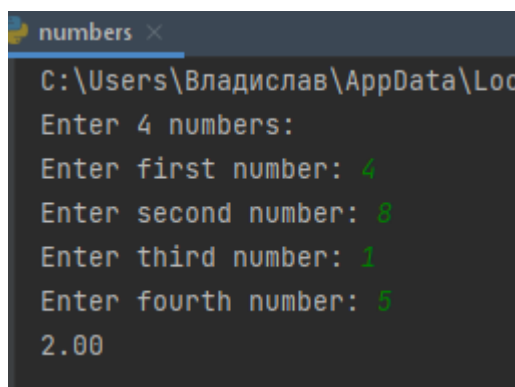
print('Enter 4 numbers: ')
num_1 = float(input("Enter first number: "))
num_2 = float(input("Enter second number: "))
num_3 = float(input("Enter third number: "))
num_4 = float(input("Enter fourth number: "))

s1 = num_1 + num_2
s2 = num_3 + num_4
result = s1 / s2

print("%.2f" % result)

```

Рис. 5 Код для решения задачи



```

numbers x
C:\Users\Владислав\AppData\Loc
Enter 4 numbers:
Enter first number: 4
Enter second number: 8
Enter third number: 1
Enter fourth number: 5
2.00

```

Рис. 6 Результат выполнения кода

5. Приступил к выполнению индивидуального задания.

### Вариант 19

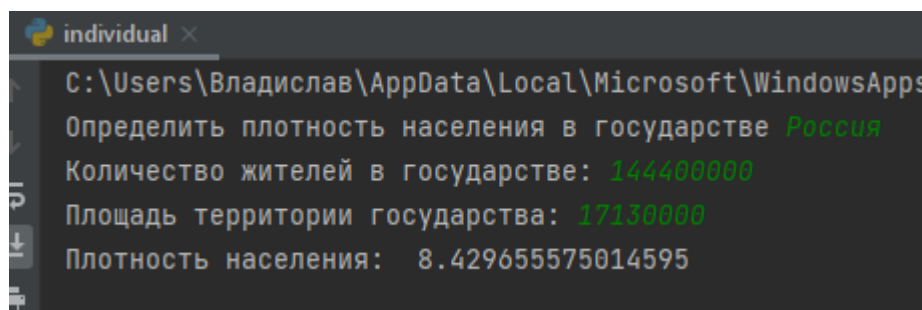
Условие: Известны количество жителей в государстве и площадь его территории. Определить плотность населения в этом государстве.

```

1 input("Определить плотность населения в государстве ")
2 population = int(input("Количество жителей в государстве: "))
3 area = int(input("Площадь территории государства: "))
4
5 density = population / area
6
7 print("Плотность населения: ", density)

```

Рис. 7 Вычисление плотности населения



```
individual x
C:\Users\Владислав\AppData\Local\Microsoft\WindowsApps
Определить плотность населения в государстве Россия
Количество жителей в государстве: 144400000
Площадь территории государства: 17130000
Плотность населения: 8.429655575014595
```

Рис. 8 Результат вычисления

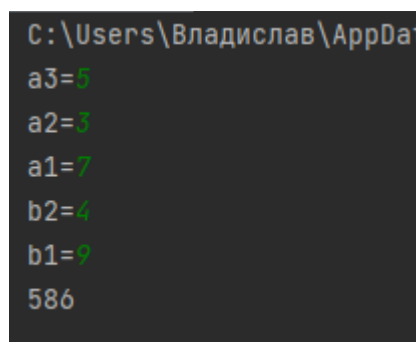
6. Приступил к выполнению задания повышенной сложности.

### Вариант 3

Условие: Даны цифры двух десятичных целых чисел: трехзначного  $a_3a_2a_1$  и двузначного  $b_2b_1$ , где  $a_1$  и  $b_1$  – число единиц,  $a_2$  и  $b_2$  – число десятков,  $a_3$  – число сотен. Получить цифры числа, равного сумме заданных чисел (известно, что это число трехзначное). Числа-слагаемые и число-результат не определять; условный оператор не использовать.

```
print((100*int(input('a3='))+10*int(input('a2='))+int(input('a1=')))+(10*int(input('b2='))+int(input('b1='))))
```

Рис. 9 Код к заданию



```
C:\Users\Владислав\AppData
a3=5
a2=3
a1=7
b2=4
b1=9
586
```

Рис. 10 Результат работы кода

### Контрольные вопросы.

1. Опишите основные этапы установки Python в Windows и Linux.

Порядок установки на Windows.

- Запустите скачанный установочный файл.
- Выберете способ установки.
- Отметьте необходимые опции установки.
- Выберете место установки (доступно при выборе Customize installation)
- После успешной установки вас ждет следующее сообщение.

Порядок установки на Linux.

Чаще всего интерпретатор Python уже входит в состав дистрибутива.

2. В чем отличие пакета Anaconda от пакета Python, скачиваемого с официального сайта?

Этот пакет включает в себя интерпретатор языка Python (есть версии 2 и 3), набор наиболее часто используемых библиотек и удобную среду разработки и исполнения, запускаемую в браузере. Основное различие между Anaconda и программированием на Python заключается в том, что Anaconda является дистрибутивом языков программирования таких как Python и R для науки о данных и машинного обучения, а Python — это язык программирования высокого уровня общего назначения.

3. Как осуществить проверку работоспособности пакета Anaconda?

Для выполнения проверки работоспособности Anaconda необходимо вначале запустить командный процессор с поддержкой виртуальных окружений Anaconda. В Windows это можно сделать выбрав следующий пункт

главного меню системы Пуск Anaconda3 (64-bit) Anaconda Prompt. В появившейся командной строке необходимо ввести:

```
> jupyter notebook
```

В результате чего отобразится процесс загрузки веб-среды Jupyter Notebook.

4. Как задать используемый интерпретатор языка Python в IDE PyCharm?

При запуске программы и создании проекта будет предложено выбрать интерпретатор языка.

5. Как осуществить запуск программы с помощью IDE PyCharm?

Нажать правой кнопки мыши на рабочее поле программы и нажать “Run”.

6. В чем суть интерактивного и пакетного режимов работы Python?

В интерактивном режиме можно писать код и сразу запускать его, в пакетном можно запускать отдельный файл с кодом.

7. Почему язык программирования Python называется языком динамической типизации?

Если достаточно формально подходить к вопросу о типизации языка Python, то можно сказать, что он относится к языкам с неявной сильной динамической типизацией. Неявная типизация означает, что при объявлении переменной вам не нужно указывать её тип, при явной – это делать необходимо.

Также языки бывают с динамической и статической типизацией. В первом случае тип переменной определяется непосредственно при выполнении программы, во втором – на этапе компиляции.

Сильная типизация не позволяет производить операции в выражениях с данными различных типов, слабая – позволяет. В языках с сильной типизацией

вы не можете складывать, например, строки и числа, нужно все приводить к одному типу.

## 8. Какие существуют основные типы в языке программирования Python?

В Python типы данных можно разделить на встроенные в интерпретатор (built-in) и не встроенные, которые можно использовать при импортировании соответствующих модулей.

К основным встроенным типам относятся:

а. None (неопределенное значение переменной)

б. Логические переменные (Boolean Type) в. Числа (Numeric Type)

– int – целое число

– float – число с плавающей точкой

– complex – комплексное число г. Списки (Sequence Type)

– list – список

– tuple – кортеж

– range – диапазон

д. Строки (Text Sequence Type)

– str

е. Бинарные списки (Binary Sequence Types)

– bytes – байты

– bytearray – массивы байт

– memoryview – специальные объекты для доступа к внутренним данным объекта через protocol buffer

ж. Множества (Set Types)

– set – множество

– frozenset – неизменяемое множество з. Словари (Mapping Types)

– dict – словарь

## 9. Как создаются объекты в памяти? Каково их устройство? В чем заключается процесс объявления новых переменных, и работа операции присваивания?

Для того, чтобы объявить и сразу инициализировать переменную необходимо написать её имя, потом поставить знак равенства и значение, с

которым эта переменная будет создана.

Целочисленное значение 5 в рамках языка Python по сути своей является объектом. Объект, в данном случае – это абстракция для представления данных, данные – это числа, списки, строки и т.п. При этом, под данными следует понимать, как непосредственно сами объекты, так и отношения между

ними (об этом чуть позже). Каждый объект имеет три атрибута – это идентификатор, значение и тип. Идентификатор – это уникальный признак объекта, позволяющий отличать объекты друг от друга, а значение – непосредственно информация, хранящаяся в памяти, которой управляет интерпретатор.

#### 10. Как получить список ключевых слов в Python?

Список ключевых слов можно получить непосредственно в программе, для этого нужно подключить модуль keyword и воспользоваться командой keyword.kwlist

#### 11. Каково назначение функций id() и type()?

Для того, чтобы посмотреть на объект с каким идентификатором ссылается данная переменная, можно использовать функцию id(). Тип переменной можно определить с помощью функции type().

#### 12. Что такое изменяемые и неизменяемые типы в Python.

В Python существуют изменяемые и неизменяемые типы.

К неизменяемым (immutable) типам относятся: целые числа (int), числа с плавающей точкой (float), комплексные числа (complex), логические переменные (bool), кортежи (tuple), строки (str) и неизменяемые множества (frozen set).

К изменяемым (mutable) типам относятся: списки (list), множества (set), словари (dict).

При создании переменной, вначале создается объект, который имеет уникальный идентификатор, тип и значение, после этого переменная может ссылаться на созданный объект. Неизменяемость типа данных означает, что созданный объект больше не изменяется. Если тип данных изменяемый, то можно менять значение объекта.

#### 13. Чем отличаются операции деления и целочисленного деления?

Деление (/) выдает результат, которой получается при обычном

делении, то есть целую и дробную часть, целочисленное деление (`//`) выводит только целую часть.

14. Какие имеются средства в языке Python для работы с комплексными числами?

Для создания комплексного числа можно использовать функцию `complex(a, b)`, в которую, в качестве первого аргумента, передается действительная часть, в качестве второго – мнимая. Либо записать число в виде `a + bj`. Комплексные числа можно складывать, вычитать, умножать, делить и возводить в степень. У комплексного числа можно извлечь действительную и мнимую части

15. Каково назначение и основные функции библиотеки (модуля) `math`?

По аналогии с модулем `math` изучите самостоятельно назначение и основные функции модуля `cmath`.

В стандартную поставку Python входит библиотека `math`, в которой содержится большое количество часто используемых математических функций. Для работы с данным модулем его предварительно нужно импортировать

16. Каково назначение именованных параметров `sep` и `end` в функции `print()`?

В `print()` предусмотрены дополнительные параметры. Например, через параметр `sep` можно указать отличный от пробела разделитель строк. Параметр `end` позволяет указывать, что делать, после вывода строки. По умолчанию происходит переход на новую строку. Однако это действие можно отменить, указав любой другой символ или строку.

17. Каково назначение метода `format()`?

Какие еще существуют средства для форматирования строк в Python?

Примечание: в дополнение к рассмотренным средствам изучите самостоятельно работу с f-строками в Python.

Посмотрим на метод `format()`:

```
>>> print("This is a {0}. It's {1}.".format("ball", "red"))
```

This is a ball. It's red  
В строке в фигурных скобках указаны номера данных, которые будут сюда подставлены. Далее к строке применяется метод `format()`. В его скобках указываются сами данные (можно использовать переменные). На нулевое местоподставится первый аргумент метода `format()`, на место с номером 1 – второй и т. д.

18. Каким образом осуществить ввод с консоли значения



целочисленной и вещественной переменной в языке Python?

Если ввести число, функция `input()` все равно вернет его строковое представление. Необходимо использовать функции преобразования типов. В данном случае с помощью функций `int()` и `float()` строковые значения переменных преобразуются соответственно в целое число и вещественное число. После этого новые численные значения присваиваются тем же переменным.

**Вывод:** в процессе выполнения лабораторной работы были исследованы процессы установки и базовых возможностей языка Python версии 3.