

**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**

Выполнил студент группы ИВТ-б-о-20-1  
Павленко М.С. « » \_\_\_\_\_ 20\_\_ г.  
Подпись студента \_\_\_\_\_  
Работа защищена « » \_\_\_\_\_ 20\_\_ г.  
Проверила Воронкин Р.А. \_\_\_\_\_

(подпись)

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

Ссылка на репозиторий: <https://github.com/0Roni0/3-lab.git>

1. Создал репозиторий и клонировал его в git:

```
79624@DESKTOP-BUV2U6A MINGW64 ~/1-lab (main)
$ git clone https://github.com/0Roni0/3-lab.git
Cloning into '3-lab'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), done.
```

Рисунок 1. Клонирование репозитория.

2. Первое задание:

Написал код:

```
name = input("What is your name?")
old = input("How old are you?")
live = input("Where are you live?")
print('This is ', name)
print('He is', old)
print('He live in', live)
```

What is your name?Mixail  
How old are you?18  
Where are you live?Stavropol  
This is Mixail  
He is 18  
He live in Stavropol  
✚

Рисунок 2. Код для задания.

3. Второго задания.

```
Решите пример: 4 * 100 - 54=?' )
input('Выше решение:')
100-54;
Ваш ответ:',otvet,'Правильный ответ:',prav
```

Решите пример: 4 \* 100 - 54=?  
Выше решение:346  
Ваш ответ: 346 Правильный ответ: 346  
✚

Рисунок 3. Код задания.

Третье задание.

```
print('Введите 4 числа')
num1=float(input('1:'))
num2=float(input('2:'))
num3=float(input('3:'))
num4=float(input('4:'))
sum1=num1+num2
sum2=num3+num4
res=sum1/sum2
print("%.2f"%res)
```

Введите 4 числа  
1:5  
2:5  
3:6  
4:3  
1.11  
✚

Рисунок 4. Код задания.

Выполнение индивидуального задания:

Вариант – 11:

Известна стоимость 1 кг конфет, печенья и яблок. Найти стоимость всей покупки, если

купили кг конфет, у кг печенья и кг яблок.

Код программы:

```
import math
PRINT ('1 kg konfet = ')
INPUT costA
PRINT ('1 kg pieczienia = ');
INPUT costB
PRINT ('1 kg yablok = ')
INPUT costC
PRINT ('X = ')
INPUT x
PRINT ('Y = ')
INPUT y
PRINT ('Z = ')
INPUT z
PRINT ('ALL COST = ');
PRINT ('costA * x + costB * y + costC * z')
```

```

79624@DESKTOP-BUV2U6A MINGW64 ~/1-lab/3-lab (main)
$ git add .

79624@DESKTOP-BUV2U6A MINGW64 ~/1-lab/3-lab (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   .gitignore
        new file:   "1 \320\267\320\260\320\264\320\260\320\275\320\270\320\265.py"
        new file:   "2 \320\267\320\260\320\264\320\260\320\275\320\270\320\265.py"
        new file:   "3 \320\267\320\260\320\264\320\260\320\275\320\270\320\265.py"
        modified:   LICENSE
        modified:   README.md
        new file:   "\320\270\320\267.py"

79624@DESKTOP-BUV2U6A MINGW64 ~/1-lab/3-lab (main)
$ git commit -m "lab"
[main 79f1d70] lab
7 files changed, 46 insertions(+), 2 deletions(-)
create mode 100644 "1 \320\267\320\260\320\264\320\260\320\275\320\270\320\265.py"
create mode 100644 "2 \320\267\320\260\320\264\320\260\320\275\320\270\320\265.py"
create mode 100644 "3 \320\267\320\260\320\264\320\260\320\275\320\270\320\265.py"
create mode 100644 "\320\270\320\267.py"

79624@DESKTOP-BUV2U6A MINGW64 ~/1-lab/3-lab (main)
$ git push
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 16 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (9/9), 1.24 KiB | 1.24 MiB/s, done.
Total 9 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/0Roni0/3-lab.git
   5b62427..79f1d70  main -> main

79624@DESKTOP-BUV2U6A MINGW64 ~/1-lab/3-lab (main)

```

Контрольные вопросы:

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

Порядок установки на Windows.

- 1) Запустите скачанный установочный файл.
- 2) Выберите способ установки.
- 3) Отметьте необходимые опции установки (доступно при выборе Customize installation)

4) Выберите место установки (доступно при выборе Customize installation)

5) После успешной установки вас ждет следующее сообщение.  
Порядок установки на Linux.

Чаще всего интерпретатор Python уже входит в состав дистрибутива.

Это можно проверить набрав в терминале: `$ python` или `$ python3`.

Если у вас, при попытке запустить Python, выдается сообщение о том, что он не установлен, или установлен, но не тот, что вы хотите, то у вас есть два пути: а) собрать Python из исходников; б) взять из репозитория. Для установки из репозитория в Ubuntu воспользуйтесь командой: `$ sudo apt-get install python3`.

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:

- 1) None (неопределенное значение переменной)
- 2) Логические переменные (Boolean Type)
- 3) Числа (Numeric Type)
  - int – целое число
  - float – число с плавающей точкой
  - complex – комплексное число
- 4) Списки (Sequence Type)
  - list – список

- tuple – кортеж
- range – диапазон
- 5) Строки (Text Sequence Type)
- str
- 6) Бинарные списки (Binary Sequence Types)
- bytes – байты
- bytearray – массивы байт
- memoryview – специальные объекты для доступа к

внутренним

данным объекта через protocol buffer

- 7) Множества (Set Types)
- set – множество
- frozenset – неизменяемое множество

- 8) 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. Имя переменной не должно совпадать с ключевыми словами интерпретатора Python.

- 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?

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`).

13. Чем отличаются операции деления и целочисленного деления?

Деление (`/`) выдает результат, который 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 smath.

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 именных 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, будут которые

сюда подставлены. Далее к строке применяется метод `format()`.

В его скобках

указываются сами данные (можно использовать переменные).

На нулевое

место подставится первый аргумент метода `format()`, на место с номером 1 – второй и т. д.

«Форматированные строковые литералы», `f-strings` являются строковыми литералами с «`f`» в начале и фигурными скобками,

содержащими

выражения, которые в дальнейшем будут заменены своими значениями.

Выражения оцениваются по мере выполнения и затем формируются при

```
помощи протокола_____format__.name = "Eric"
```

```
age = 74
```

```
print(f "Hello, {name}. You are {age}.")# Вывод: 'Hello, Eric. You are 74.'
```

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. 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 ходе выполнения лабораторной работы исследован процесс установки базовых возможностей языка Python 3.