

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

**Кафедра инфокоммуникаций**

**Основы кроссплатформенного программирования**

**Отчет по лабораторной работе №2**

**«Основы языка Python3»**

Выполнил студент группы

ИТС-б-о-20-1(2)

Аблаев Д.К. « » \_\_\_\_\_ 20\_\_ г.

Подпись студента \_\_\_\_\_

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

Проверил доцент

Кафедры инфокоммуникаций, старший  
преподаватель

Воронкин Р.А.

\_\_\_\_\_  
(подпись)

Ставрополь 2021

**Цель работы:** исследование процесса установки и базовых возможностей языка Python3.

Репозиторий: <https://github.com/AblaevDaniil/LR2>

**Выполнение работы:**

1. Создадим общедоступный репозиторий на GitHub, решим задачи с помощью языка программирования Python3, отправим их на GitHub.

**Задача 1.**

7. Напишите программу (файл *user.py*), которая запрашивала бы у пользователя:

- его имя (например, "What is your name?")
- возраст ("How old are you?")
- место жительства ("Where are you live?")

После этого выводила бы три строки:

```
"This is `имя`"  
"It is `возраст`"  
"(S)he live in `место_жительства`"
```

Рисунок 1. Задача 1

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

```
What is your name?  
Daniil  
How old are you?  
18  
Where are you live?  
Stavropol  
This is Daniil  
It is 18  
(S)he live in Stavropol
```

**Задача 2.**

9. Напишите программу (файл *arithmetic.py*), которая предлагала бы пользователю решить пример  $4 * 100 - 54$ . Потом выводила бы на экран правильный ответ и ответ пользователя. Подумайте, нужно ли здесь преобразовывать строку в число.

```
1 a = int(input())
2 print("4*100-54", "=", a)
3 print("ваш ответ:", a)
4 print("Правильный ответ: 346")
```

345

4\*100-54 = 345

ваш ответ: 345

Правильный ответ: 346

### Задача 3.

10. Запросите у пользователя четыре числа (файл *numbers.py*). Отдельно сложите первые два и отдельно вторые два. Разделите первую сумму на вторую. Выведите результат на экран так, чтобы ответ содержал две цифры после запятой.

```
1 a = int(input())
2 b = int(input())
3 c = int(input())
4 d = int(input())
5 sum1 = a+b
6 sum2 = c+d
7 m = sum1/sum2
8 print("%.2f" %m)
9
```

23

30

45

15

0.88

### Задание 4.

2. Даны стороны прямоугольника. Найти его периметр и длину диагонали.

```

1  a = int(input())
2  b = int(input())
3  p = (a+b)*2
4  d = (a**2+b**2)**0.5
5  print("Периметр прямоугольника", p)
6  print("Длина диагонали прямоугольника", d)
7

```

```

3
4
Периметр прямоугольника 14
Длина диагонали прямоугольника 5.0

```

**Вывод:** Исследован процесс установки и базовые возможности языка Python3, написано несколько задач.

Контрольные вопросы:

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

Основные этапы установки Python в Windows и Linux:

- 1) Скачивание дистрибутива с сайта python.org;
- 2) Запуск установочного файла;
- 3) Выбор необходимых опций установки;
- 4) Указание пути установки;
- 5) Установка Python.

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

Пакет Anaconda содержит в себе интерпретатор языка Python 2 и 3 версии, набор наиболее часто используемых библиотек и удобную среду разработки и исполнения.

3. Как осуществить проверку работоспособности пакета Anaconda?

Для проверки работоспособности пакета Anaconda нужно запустить программу Anaconda Prompt, после в появившейся командной строке ввести «jupyter notebook». После этих действий отобразится процесс загрузки веб-среды Jupyter Notebook. В открывшейся среде нужно создать новый ноутбук, после в поле для кода ввести пробный код, например «print(“Hello, World!”)».

Если после запуска кода на экране появилась надпись «Hello, World!», то Anaconda установлен правильно.

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

Используемый интерпретатор языка Python в IDE PyCharm задается при создании нового проекта в соответствующей строке.

5. Как осуществить запуск программы с помощью IDE PyCharm?

Для запуска программы с помощью IDE PyCharm нужно открыть код программы, после чего в правом верхнем углу нажать кнопку «Run».

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

В интерактивном режиме работы Python ожидает ввода команд пользователя. При вводе команды интерпретатор выполнит строку и отобразит строкой ниже результат своей работы.

В пакетном режиме работы Python будет только выполнять уже написанный код. Для этого нужно набрать в командной строке `python «название файла».py`.

7. Почему язык программирования Python называется языком динамической типизации?

Язык Python называется языком динамической типизации потому, что тип переменной определяется непосредственно при выполнении программы, а не на этапе компиляции, как в языках статической типизации.

8. Какие существуют основные типы в языке программирования Python?

Основные типы данных:

- 1) None (неопределенное значение переменной);
- 2) логические переменные;
- 3) числа;
- 4) списки; строки;
- 5) бинарные списки (байты, массивы байт);
- 6) множества;

7) словари.

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

Объект – это абстракция для представления данных, данные – это числа, списки, строки и т.д. Для создания объекта нужно написать его имя, потом поставить знак равенства и значение, с которым объект будет создан.

10. Как получить список ключевых слов в Python?

Для получения списка ключевых слов в Python нужно подключить модуль keyword и воспользоваться командой keyword.kwlist.

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

Функции id() и type() задают идентификатор объекту, тип переменной.

12. Что такое изменяемые и неизменяемые типы в Python.

Изменяемые типы данных – типы данных, которые могут изменяться в процессе выполнения кода программы. К ним относятся списки (list), множества (set) и словари (dict).

Неизменяемые типы данных – типы данных, неизменяемые в процессе выполнения кода программы. К ним относятся целые числа (int), числа с плавающей точкой (float), комплексные числа (complex), логические переменные (bool), кортежи (tuple), строки (str) и неизменяемые множества (frozen set).

13. Чем отличаются операции деления и целочисленного деления?

Деление возвращает частное с его дробной частью (при наличии).

Целочисленное деление возвращает целую часть частного, а дробная часть отбрасывается.

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

Для работы с комплексными числами используются функции complex(a, b), x.real, x.imag, x.conjugate().

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

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

Библиотека `math` содержит в себе большое количество часто используемых математических функций, например `math.celi(x)`, `math.fabs(x)`, `math.factorial(x)` и т.д.

Основные функции библиотеки: `math.celi(x)`, `math.fabs(x)`, `math.factorial(x)`, `math.floor(x)`, `math.exp(x)`, `math.log2(x)`, `math.log10(x)`, `math.log(x[, base])`, `math.pow(x,y)`, `math.sqrt(x)`, `math.sin(x)`, `math.cos(x)`, `math.tan(x)`, `math.acos(x)`, `math.asin(x)`, `math.atan(x)`, `math.pi`, `math.e`.

Модуль `cmath` предоставляет функции для работы с комплексными числами.

Основные функции модуля `cmath`: `cmath.phase(x)`, `cmath.polar(x)`, `cmath.rect(x)`, `cmath.exp(x)`, `cmath.log(x[, base])`, `cmath.acosh(x)`, `cmath.asinh(x)`, `cmath.atanh(x)`.

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

Параметр `sep` указывает отличный от пробела разделитель строк.

Параметр `end` указывает, что делать после вывода строки (поставить знак, сделать перенос на несколько строк).

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.

Метод `format()` применяется к строке и позволяет подставлять типы данных в неё (`%s`, `%d`, `%f`), указывать количество знаков после запятой для чисел (`%1f`).

Метод `format()` применяется к строке и позволяет подставлять типы данных в неё (`%s`, `%d`, `%f`), указывать количество знаков после запятой для чисел (`%lf`).

18. Каким образом осуществить ввод с консоли значения целочисленной и вещественной переменной в языке Python?

Ввод с консоли значения переменной производится при помощи функции `input()`.