

**РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**Отчет по лабораторной работе № 2  
по дисциплине «Программирование на Python»**

Выполнил студент группы ИВТ-б-о-24-1:  
Хубиев Роберт Эльбрусович  
«28» октября 2025г.

Подпись студента \_\_\_\_\_ Хубиев \_\_\_\_\_  
Работа защищена « » \_\_\_\_\_ 2025г.

Проверил Воронкин Р.А. \_\_\_\_\_  
(подпись)

Ставрополь 2025

**Ссылка на репозиторий:** <https://github.com/truebobsuncle/LAR222>

**Тема:** основы языка Python.

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

### **Порядок выполнения работы:**

- 1)** Создали общедоступный репозиторий
- 2)** Выполнили клонирование репозитория
- 3)** Дополнили .gitignore необходимыми правилами для работы с IDE PyCharm:

```
# PyCharm
.idea/
*.iws
*.iml
*.ipr
```

Рисунок 1. Правила

- 4)** Создали ветку develop и перешли в неё:

```
(base) C:\Users\Роберт\PycharmProjects\pythonProject>git checkout -b develop
Switched to a new branch 'develop'
```

Рисунок 2. Создание ветки

- 5)** Написали программу user.py:

```
name = input("What is your name? ")  
age = input("How old are you? ")  
city = input("Where are you live? ")  
  
print(f"This is {name}")  
print(f"It is {age}")  
print(f"he live in {city}")
```

Run user  
How old are you? 18  
Where are you live? Stavropol  
This is Bob  
It is 18  
he live in Stavropol  
Process finished with exit code 0

Рисунок 3. Результат выполнения программы

## 6) Написали программу arithmetic.py:

```
user_answer = input("Решите пример: 4 * 100 - 54 = ")  
correct_answer = 4 * 100 - 54 # 400 - 54 = 346  
  
print(f"Правильный ответ: {correct_answer}")  
print(f"Ваш ответ: {user_answer}")
```

Run arithmetic  
C:\anaconda3\python.exe C:/Users/Роберт/PycharmProjects/pythonProject/arithmetic.py  
Решите пример: 4 \* 100 - 54 = 3212  
Правильный ответ: 346  
Ваш ответ: 3212  
Process finished with exit code 0

Рисунок 4. Результат выполнения программы

## 7) Написали программу numbers.py:

```
1 num1 = float(input("Введите первое число: "))
2 num2 = float(input("Введите второе число: "))
3 num3 = float(input("Введите третье число: "))
4 num4 = float(input("Введите четвертое число: "))
5
6 sum1 = num1 + num2
7 sum2 = num3 + num4
8
9 result = sum1 / sum2
10 print(f"Результат: {result:.2f}")
```

Run numbers

Введите первое число: 2  
Введите второе число: 3  
Введите третье число: 4  
Введите четвертое число: 5  
Результат: 0.56

Process finished with exit code 0

Рисунок 5. Результат выполнения программы

### 8) Написали программу individual.py:

**Задание:** даны катеты прямоугольного треугольника. Найти его периметр.

```
1 import math
2 print("Введите катеты прямоугольного треугольника:")
3 a = float(input("Первый катет: "))
4 b = float(input("Второй катет: "))
5 c = math.sqrt(a**2 + b**2)
6 perimeter = a + b + c
7 print(f"Гипотенуза: {c:.2f}")
8 print(f"Периметр треугольника: {perimeter:.2f}")
```

Run individual

Введите катеты прямоугольного треугольника:  
Первый катет: 3  
Второй катет: 4  
Гипотенуза: 5.00  
Периметр треугольника: 12.00

Process finished with exit code 0

Рисунок 6. Результат выполнения программы

## **Контрольные вопросы:**

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

Windows:

- 1)Скачать установщик с официального сайта [python.org](http://python.org)
- 2)Запустить .exe файл и отметить галочку "Add Python to PATH"
- 3)Выбрать тип установки (рекомендуется обычная)
- 4)Завершить установку и проверить через команду python --version

Linux:

- 1)Обновить пакеты: sudo apt update
- 2)Установить Python: sudo apt install python3
- 3)Проверить установку: python3 --version

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

Anaconda — это дистрибутив, включающий:

- 1) Интерпретатор Python
- 2) Предустановленные библиотеки для Data Science (NumPy, Pandas, Matplotlib, Jupyter)
- 3) Менеджер пакетов Conda для управления окружениями
- 4) Графический интерфейс Anaconda Navigator

Официальный Python — только интерпретатор и базовые библиотеки.

### **3. Как осуществить проверку работоспособности пакета Anaconda?**

1) Открыть командную строку или Anaconda Prompt

2) Ввести команды: bash, conda –version, python –version, jupyter --version

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

1) Открыть Settings (Ctrl+Alt+S)

2) Перейти в раздел: Project -- Python Interpreter

3) Нажать на шестерёнку -- Add Interpreter

4) Выбрать путь к интерпретатору (например, C:\anaconda3\python.exe)

### **5. Как осуществить запуск программы с помощью IDE PyCharm?**

1) Открыть файл с кодом (.py)

2) Нажать сочетание клавиш Ctrl+Shift+F10

3) Или правой кнопкой по файлу -- Run

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

**1)Интерактивный режим** — выполнение кода построчно с немедленным результатом (например, в Jupyter Notebook).

**2)Пакетный режим** — запуск готовой программы из файла целиком

### **7. Почему язык программирования Python называется языком динамической типизации?**

Типы переменных определяются автоматически во время выполнения программы, а не при компиляции.

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

- 1) int (целые числа)
- 2) float (вещественные числа)
- 3) str (строки)
- 4) bool (логический тип)
- 5) list (списки)
- 6) tuple (кортежи)
- 7) dict (словари)
- 8) set (множества)

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

При создании переменной:

- 1) В памяти выделяется место для объекта
- 2) Переменная становится ссылкой на этот объект
- 3) Оператор присваивания (=) связывает имя переменной с объектом

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

```
python
import keyword
print(keyword.kwlist)
```

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

- 1) id() — возвращает уникальный идентификатор объекта в памяти
- 2) type() — возвращает тип объекта

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

- 1) Неизменяемые: int, float, str, tuple — нельзя изменить после создания.
- 2) Изменяемые: list, dict, set — можно изменять содержимое.

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

- 1) Обычное деление: / — возвращает float ( $7 / 2 = 3.5$ )
- 2) Целочисленное деление: // — возвращает int ( $7 // 2 = 3$ )

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

- 1) print(z.real) # действительная часть
- 2) print(z.imag) # мнимая часть

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

math — математические функции:

- math.sqrt() — квадратный корень

- `math.sin()`, `math.cos()` — тригонометрия
- `math.pi`, `math.e` — константы

`cmath` — аналогичные функции для комплексных чисел.

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

- 1) `sep` — разделитель между аргументами (по умолчанию пробел)
- 2) `end` — что выводить в конце (по умолчанию перевод строки)

**17. Каково назначение метода `format()`? Какие еще существуют средства для форматирования строк в Python?**

`format()` — подстановка значений в строку:

```
python "{ } + { } = { }".format(2, 3, 5)
```

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

```
x = int(input("Введите целое число: "))  
y = float(input("Введите вещественное число: "))
```

**Вывод:** Освоены основы программирования на Python, работа с IDE PyCharm.

