

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт перспективной инженерии  
Департамент цифровых, робототехнических систем и электроники

**ОТЧЕТ**  
**ПО ЛАБОРАТОРНОЙ РАБОТЕ №4**  
**дисциплины**  
**«Основы кроссплатформенного программирования»**  
**Вариант 4**

Выполнил:  
Борцов Богдан Михайлович  
2 курс, группа ИТС-б-о-23-1,  
11.03.02  
«Инфокоммуникационные  
технологии и системы связи»,  
направленность (профиль)  
«Инфокоммуникационные  
системы и сети»,  
очная форма обучения

---

(подпись)

Проверил:  
Доцент департамента цифровых,  
робототехнических систем и  
электроники  
Воронкин Р.А.

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2024 г.

## Лабораторная работа 1.3 «Основы языка Python»

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

### Ход работы:

1. Создан общедоступный репозиторий на GitHub
2. Выполнил задание «user.py»:

```
name = input("What is your name? ")
age = input("How old are you? ")
location = input("Where do you live? ")

print(f"This is {name}")
print(f"It is {age}")
print(f"(S)he live in {location}")
```

Рисунок 2. Программа задания «user»

3. Выполнил задание «arithmetic»:

```
print("Solve the example: 4 * 100 - 54")

user_answer = input("Your answer: ")

correct_answer = 4 * 100 - 54

print(f"The correct answer is: {correct_answer}")
print(f"Your answer was: {user_answer}")
```

Рисунок 3. Программа задания «arithmetic»

4. Выполнил задание «numbers»

```

num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))
num3 = float(input("Enter the third number: "))
num4 = float(input("Enter the fourth number: "))

sum1 = num1 + num2
sum2 = num3 + num4

result = sum1 / sum2

print(f"The result is: {result:.2f}")

```

Рисунок 4. Программа задания «numbers»

5. Выполнил индивидуальное задание:

4. Даны два числа. Найти их сумму, разность, произведение, а также частное от деления первого числа на второе.

Рисунок 5. Индивидуальное задание в соответствии с вариантом

```

# Ввод двух чисел
num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))

# Вычисления
sum_result = num1 + num2
difference = num1 - num2
product = num1 * num2

# Проверка деления на ноль
if num2 != 0:
    division = num1 / num2
else:
    division = "undefined (division by zero)"

# Вывод результатов
print(f"Sum: {sum_result}")
print(f"Difference: {difference}")
print(f"Product: {product}")
print(f"Division: {division}")

```

## Рисунок 6. Программа для индивидуального задания

6. Закоммитил изменения
7. Выполнил слияние ветки разработки с основной веткой
8. Запустил изменения на удаленный сервер:

### Ответы на контрольные вопросы:

#### 1. Основные этапы установки Python в Windows и Linux:

*Windows:* Скачайте установочный файл с официального сайта Python, запустите его и следуйте инструкциям, не забыв отметить опцию "Add Python to PATH".

*Linux:* Используйте пакетный менеджер вашей дистрибуции, например, `sudo apt install python3` для Ubuntu.

#### 2. Различие пакета Anaconda от пакета Python:

*Anaconda* включает в себя множество библиотек и инструментов для научных вычислений и анализа данных, таких как Jupyter Notebook и Conda.

*Официальный Python* поставляется только с основными библиотеками и инструментами.

#### 3. Проверка работоспособности пакета Anaconda:

Запустите `conda list` в командной строке, чтобы увидеть установленные пакеты.

#### 4. Задание интерпретатора Python в IDE PyCharm:

Перейдите в *File > Settings > Project: <ваш проект> > Python Interpreter* и выберите необходимый интерпретатор.

#### 5. Запуск программы с помощью IDE PyCharm:

Нажмите кнопку *Run* или используйте сочетание клавиш *Shift + F10*.

#### 6. Суть интерактивного и пакетного режимов работы Python:

*Интерактивный режим:* Прямой ввод и выполнение команд в интерактивной оболочке.

*Пакетный режим:* Выполнение команд из сохраненного файла скрипта.

#### 7. Почему Python называется языком динамической типизации:

Переменные в Python могут изменять свой тип во время выполнения программы, и тип данных определяется автоматически.

#### 8. Основные типы данных в Python:

`int`, `float`, `str`, `list`, `tuple`, `dict`, `set`, `bool`.

## 9. **Создание объектов в памяти и объявление переменных:**

Объекты создаются при присваивании значений переменным, а память под них выделяется автоматически.

## 10. **Получение списка ключевых слов в Python:**

Выполните команду `import keyword; print(keyword.kwlist)`.

## 11. **Назначение функций `id()` и `type()`:**

`id()` возвращает уникальный идентификатор объекта.

`type()` возвращает тип объекта.

## 12. **Изменяемые и неизменяемые типы в Python:**

*Изменяемые:* list, dict, set.

*Неизменяемые:* int, float, str, tuple.

## 13. **Отличие операций деления и целочисленного деления:**

`/` выполняет обычное деление.

`//` выполняет целочисленное деление.

## 14. **Средства Python для работы с комплексными числами:**

Встроенные функции и модуль `cmath`.

## 15. **Назначение и функции библиотеки `math`:**

Библиотека `math` предоставляет математические функции, такие как `sqrt`, `sin`, `cos`.

Модуль `cmath` аналогичен `math`, но для комплексных чисел.

## 16. **Назначение параметров `sep` и `end` в функции `print()`:**

`sep` задает разделитель между аргументами.

`end` задает окончание строки.

## 17. **Назначение метода `format()` и другие средства форматирования строк:**

Метод `format()` используется для форматирования строк.

Альтернативы: f-строки, оператор `%`.

## 18. **Ввод значения целочисленной и вещественной переменной с консоли:**

```
int_value = int(input("Введите целое число: "))
```

```
float_value = float(input("Введите вещественное число: "))
```

**Вывод:** в ходе работы изучены основы установки и использования Python, включая интерактивный и пакетный режимы, работу с переменными, типами данных, математическими модулями и форматированием строк. Реализованы задания для закрепления навыков программирования, выполнено слияние веток и отправка изменений на GitHub. Освоены базовые принципы использования Python для решения прикладных задач.