

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО  
ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»  
ИНСТИТУТ ЦИФРОВОГО РАЗВИТИЯ

Отчет о лабораторной работе №4 по дисциплине  
«Основы программной инженерии»

Выполнил студент  
2 курса, группы ПИЖ-б-о-20-1  
Тотубалина С.С.

Проверил:  
Доцент кафедры инфокоммуникаций,  
Воронкин Р.А.

Ставрополь, 2021 г

## Ход работы

### 1. Создание ветки “develop”

```
C:\Users\admin\lab.4>git checkout -b develop  
Switched to a new branch 'develop'
```

Рис. 1 - создание ветки “develop”

### 2. Программа user.py

```
name = input("What is your name? ")  
age = input("How old are you? ")  
place = input("Where do you live? ")  
print("This is ", name)  
print("(S)he is ", age)  
print("(S)he lives in ", place)
```

Рис. 2 – код программы user.py

```
What is your name? Svetlana  
How old are you? 19  
Where do you live? Stavropol  
This is Svetlana  
(S)he is 19  
(S)he lives in Stavropol  
  
Process finished with exit code 0
```

Рис. 3 – результат выполнения программы user.py

### 3. Программа arithmetic.py

```
print("Solve the problem: 4 * 100 - 54")  
answer = input("Your answer: ")  
print("Your answer: ", answer)  
print("The right answer: 364")
```

Рис. 4 – код программы arithmetic.py

```
Solve the problem: 4 * 100 - 54  
Your answer: 346  
Your answer: 346  
The right answer: 364  
  
Process finished with exit code 0
```

Рис. 5 – результат выполнения программы arithmetic.py

#### 4. Программа numbers.py

```
print("Input four numbers: ")
a, b, c, d, = input().split()
a = int(a) + int(b)
c = int(c) + int(d)
print("The division is: %.2f" % (a/c))
```

Рис. 6 – код программы numbers.py

```
Input four numbers:
4 2 3 1
The division is: 1.50

Process finished with exit code 0
```

Рис. 7 – результат выполнения программы numbers.py

#### 5. Программа individual.py, вариант №22

Найти площадь кольца по заданным внешнему и внутреннему радиусам.

```
import math

R = float(input("Введите значение внешнего радиуса окружности: "))
r = float(input("Введите значение внутреннего радиуса окружности: "))
SR = math.pi * (R**2)
Sr = math.pi * (r**2)
S = SR - Sr
S = float('{:.3f}'.format(S))
print('Площадь кольца = ', S)
```

Рис. 8 – код программы individual.py

```
Введите значение внешнего радиуса окружности: 15
Введите значение внутреннего радиуса окружности: 9
Площадь кольца = 452.389

Process finished with exit code 0
```

Рис. 9 – результат выполнения программы individual.py

Ответы на вопросы:

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

Windows: Для операционной системы Windows дистрибутив распространяется либо в виде исполняемого файла (с расширением exe), либо в виде архивного файла (с расширением zip). Если вы используете Windows 7, не забудьте установить Service Pack 1!

Порядок установки.

1. Запустите скачанный установочный файл.

2. Выберите способ установки.

В данном окне предлагается два варианта Install Now и Customize installation. При выборе Install Now, Python установится в папку по указанному пути. Помимо самого интерпретатора будет установлен IDLE (интегрированная среда разработки), pip (пакетный менеджер) и документация, а также будут созданы соответствующие ярлыки и установлены связи файлов, имеющие расширение .py с интерпретатором Python. Customize installation – это вариант настраиваемой установки. Опция Add python 3.5 to PATH нужна для того, чтобы появилась возможность запускать интерпретатор без указания полного пути до исполняемого файла при работе в командной строке.

3. Отметьте необходимые опции установки (доступно при выборе Customize installation)

На этом шаге нам предлагается отметить дополнения, устанавливаемые вместе с интерпретатором Python. Рекомендуется выбрать все опции.

Documentation – установка документаций. pip – установка пакетного менеджера pip.

tcl/tk and IDLE – установка интегрированной среды разработки (IDLE) и библиотеки для построения графического интерфейса (tkinter).

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

Помимо указания пути, данное окно позволяет внести дополнительные изменения в процесс установки с помощью опций:

Install for all users – Установить для всех пользователей. Если не выбрать данную опцию, то будет предложен вариант инсталляции в папку пользователя, устанавливающего интерпретатор.

Associate files with Python – Связать файлы, имеющие расширение .py, с Python. При выборе данной опции будут внесены изменения в Windows, позволяющие запускать Python скрипты по двойному щелчку мыши.

Create shortcuts for installed applications – Создать ярлыки для запуска приложений.

Add Python to environment variables – Добавить пути до интерпретатора Python в переменную PATH.

Precompile standard library – Провести прекомпиляцию стандартной библиотеки.

Последние два пункта связаны с загрузкой компонентов для отладки, их мы устанавливать не будем.

Linux: Чаще всего интерпретатор Python уже входит в состав дистрибутива. Это можно проверить, набрав в терминале «python» или «python3».

В первом случае, вы запустите Python 2 во втором – Python 3. В будущем, скорее всего, во всех дистрибутивах Linux, включающих Python, будет входить только третья версия. 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?

Для удобства запуска примеров и изучения языка Python, настоятельно рекомендуется установить на свой ПК пакет Anaconda. 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.

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»,

запустится веб-сервер и среда разработки в браузере. Создайте ноутбук для разработки, для этого нажмите на кнопку New (в правом углу окна) и в появившемся списке выберете Python. В результате будет создана новая страница в браузере с ноутбуком. Введите в первой ячейке команду «`print("Hello, World!")`» и нажмите Alt+Enter на клавиатуре. Ниже ячейки должна появиться соответствующая надпись.

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

При создании нового проекта нужно будет указать путь до него и интерпретатор.

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

После создания нового проекта нужно добавить python файл в проект.

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

Интерактивный режим – непосредственное выполнение команд одна за другой в консоли. Пакетный режим – запуск программы из файла.

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

Потому что тип переменной определяется непосредственно при выполнении программы.

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

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. 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. При инициализации переменной, на уровне интерпретатора, происходит следующее:

создается целочисленный объект 5 (можно представить, что в этот момент создается ячейка и 5 кладется в эту ячейку);

данный объект имеет некоторый идентификатор, значение: 5, и тип: целое число;

посредством оператора “=” создается ссылка между переменной b и целочисленным объектом 5 (переменная b ссылается на объект 5). 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`).

Как уже было сказано ранее, 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. Например, если мы объявим переменную `k = 15`, то будет создан объект со значением 15, типа `int` и идентификатором, который можно узнать с помощью функции `id()`.

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

При обычном делении результатом операции будет вещественное число с плавающей точкой. При целочисленном делении результатом будет целое число, показывающее количество целых чисел `b` в числе `a`, к примеру.

#### 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. Библиотека «cmath» содержит в себе функции для работы с комплексными числами.

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

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.

В строке в фигурных скобках указаны номера данных, которые будут 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.

«f-строки»: Форматирование, которое появилось в Python 3.6 (PEP 498). Этот способ похож на форматирование с помощью метода `format()`, но гибче, читабельней и быстрее. Пример:

```
>>> name = "Дмитрий"
>>> age = 25
>>> print(f"Меня зовут {name} Мне {age} лет.")
>>> Меня зовут Дмитрий. Мне 25 лет.
```

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

18) Даже, если ввести число, функция `input()` все равно вернет его строковое представление. Чтобы получить число, нужно использовать функции преобразования типов. Пример:

```
qtyOranges = int(input("Сколько апельсинов? "))
priceOrange = float(input("Цена одного апельсина? "))
sumOranges = qtyOranges * priceOrang
```