

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

Кафедра инфокоммуникаций

Лабораторная работа 3

Основы языка Python3

Выполнил студент группы ИВТ-б-о-20-1

Симанский М.Ю « » _____ 20__ г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил Воронкин Р.А. _____

(подпись)

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

Ссылка на репозиторий: <https://github.com/MaksimSimanskiy/Lab3.git>

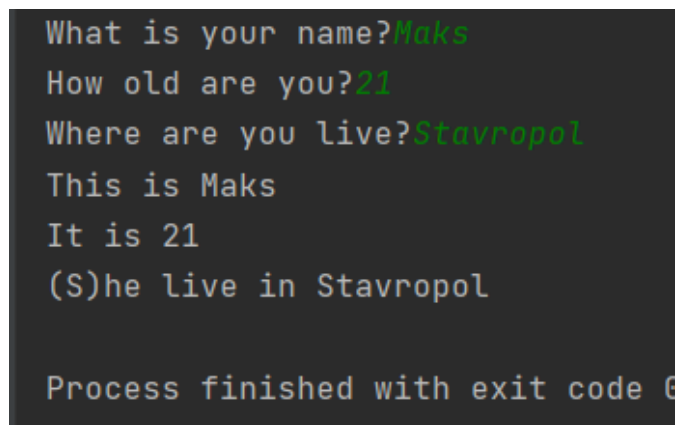
Задание 1

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?")

Код

```
a = input('What is your name?')
b = input('How old are you?')
c = input('Where are you live?')
print('This is', a)
print('It is', b)
print('(S)he live in', c )
```

Результат



```
What is your name?Maks
How old are you?21
Where are you live?Stavropol
This is Maks
It is 21
(S)he live in Stavropol

Process finished with exit code 0
```

Рисунок 1 – Результат работы

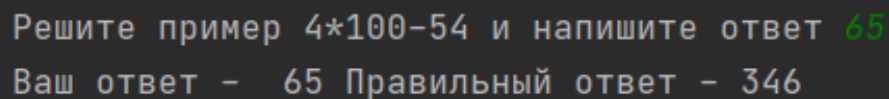
Задание 2

Напишите программу (файл arithmetic.py), которая предлагала бы пользователю решить пример $4 * 100 - 54$. Потом выводила бы на экран правильный ответ и ответ пользователя.

Код

```
offer = input('Решите пример 4*100-54 и напишите ответ ')
a = 4*100-54
print('Ваш ответ - ', offer, 'Правильный ответ -', a)
```

Результат



```
Решите пример 4*100-54 и напишите ответ 65
Ваш ответ - 65 Правильный ответ - 346
```

Рисунок 2 – Результат работы

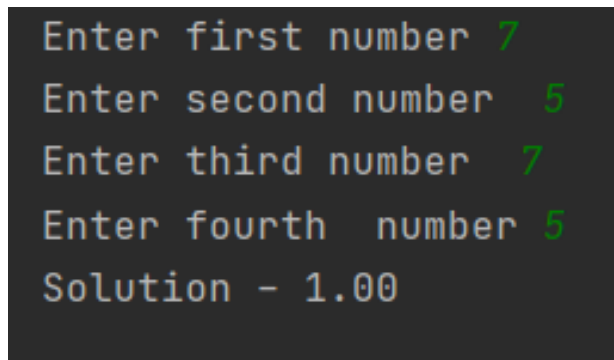
Задание 3

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.

Код

```
a = float(input('Enter first number '))
b = float(input('Enter second number '))
c = float(input('Enter third number '))
d = float(input('Enter fourth number '))
e = a + b
f = c + d
g = e / f
print("Solution - {0:.2f}".format(g))
```

Результат



```
Enter first number 7
Enter second number 5
Enter third number 7
Enter fourth number 5
Solution - 1.00
```

Рисунок 3 – Результат работы

Индивидуальное задание

17. Напишите программу, в которой вычисляется сумма, разность, произведение, частное и среднее арифметическое двух целых чисел, введенных с клавиатуры.

Например, при вводе

чисел 2 и 7 должен быть получен ответ вида:

$2+7=9$ $2-7=-5$ $2*7=14$ $2/7=0.2857142857142857$ $(2+7)/2=4.5$

Код

```
a = int(input('Enter first number '))
b = int(input('Enter second number '))
sum = (a+b)
diff = (a-b)
mult = a*b
quot = float(a/b)
avg = (a+b)/2
print(a, '+', b, '=', sum,
      a, '-', b, '=', diff,
      a, '*', b, '=', mult,
      a, '/', b, '=', quot,
      '(', a, '+', b, ')', '/', 2, '=',
      avg)
```

Результат

```

Enter first number 56
Enter second number 4
56 + 4 = 60 56 - 4 = 52 56 * 4 = 224 56 / 4 = 14.0 ( 56 + 4 ) /2= 30.0

```

Рисунок 4 – Результат работы

Задание повышенной сложности

- Даны цифры двух целых чисел: двузначного и однозначного, где – число единиц,
– число десятков. Получить цифры числа, равного сумме заданных чисел (известно, что это число двузначное). Слагаемое – двузначное число и число-результат не определять;
условный оператор не использовать.

Код

```

a1 = int(input('Units of first number a1 - '))
a2 = int(input('Tens of first number a2 - '))
b = int(input('Single number B - '))
d = (a1 + b) % 10
f = a2 + (a1 + b) // 10
print('Numbers of sum ', f, ' ', d)

```

Результат

```

Units of first number a1 - 5
Tens of first number a2 - 5
Single number B - 6
Numbers of sum 6 1

```

Рисунок 5 – Результат работы

Ответы на контрольные вопросы

1. Установка Python в Windows:

- Запустите скачанный установочный файл.
- Выберет способ установки
- Отметьте необходимые опций установки (доступно при выборе Customize installation)
- Выберите место установки (доступно при выборе Customize installation)
- Установка Python в Linux:
- Чаще всего интерпретатор Python уже входит в состав дистрибутива.

2. Пакет Anaconda 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. 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 в результате чего отобразиться процесс загрузки веб-среды Jupyter Notebook `print("Hello, World!") exit() > jupyter notebook`

4. Перейдите в Settings.

Перейдите в раздел Project Interpreter.

Нажмите кнопку "плюс" (+) и выберите параметр "Local".

Поиск интерпретатора в Linux используется в `/usr/bin/python` или `/usr/bin/pythonX.Y` где X и Y - это версия. В Windows он используется в `C:\python`.

На последнем шаге сохраните настройки, и вы настроили интерпретатор.

5. Если же у вас есть файл с исходным кодом на Python, и вы его хотите запустить, то для этого нужно в командной строке вызвать интерпретатор Python и в качестве аргумента передать ваш файл. Например, для файла с

именем test.py процедура запуска будет выглядеть так: Интерактивный режим
Откройте Python в интерактивном режиме и наберите в нем следующее: И нажмете ENTER. В ответ на это интерпретатор выполнит данную строку и отобразит строкой ниже результат своей работы. > python test.py print("Hello, World!")

6. Интерактивный режим позволяет сразу писать и запускать программы в консоли. В пакетном режиме требуется сначала создать файл.

7. В Python тип переменной определяется непосредственно при выполнении программы.

8. К основным встроенным типам относятся:

- None (неопределенное значение переменной)
- Логические переменные (Boolean Type)
- Числа (Numeric Type)
 - int – целое число
 - float – число с плавающей точкой
 - complex – комплексное число
- Списки (Sequence Type)
 - Строки (Text Sequence Type) 1. str
 - Бинарные списки (Binary Sequence Types)
 - memoryview – специальные объекты для доступа к внутренним данным объекта через protocol buffer
- Множества (Set Types)
- Словари (Mapping Types) 1. dict – словарь

9. Для того, чтобы объявить и сразу инициализировать переменную необходимо написать её имя, потом поставить знак равенства и значение, с которым эта переменная будет создана. Например строка: объявляет переменную b и присваивает ей значение 5. 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

10. Воспользоваться командой `keyword.kwlist`.

11. 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. В Python существуют изменяемые и неизменяемые типы. 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.

13. **Целочисленное деление отличается от** обычной **операции деления** тем, что возвращает целую часть частного, а дробная часть отбрасывается.

14. 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. В стандартную поставку Python входит библиотека `math`, в которой содержится большое количество часто используемых математических функций.

16. 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. Форматирование может выполняться в так называемом старом стиле или с помощью строкового метода `format`. Старый стиль также называют Си-стилем, так как он схож с тем, как происходит вывод на экран в языке C. Рассмотрим пример: Здесь вместо трех комбинаций символов `%s`, `%d`, `%f` подставляются значения переменных `rupil`, `old`, `grade`. Буквы `s`, `d`, `f` обозначают типы данных – строку, целое число, вещественное число. Если бы требовалось подставить три строки, то во всех случаях использовалось бы сочетание `%s`.

18. `qtyOranges = int(input("Сколько апельсинов? "))`
`priceOrange = float(input("Цена одного апельсина? "))`