

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

Кафедра инфокоммуникаций

«Основы языка Python»

Отчет по лабораторной работе № 2.1

**по дисциплине «Основы кроссплатформенного
программирования»**

Выполнил студент группы ИВТ-б-о-21-1

Шайдеров Дмитрий Викторович.

«13 » апреля 2022г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил Воронкин Р.А. _____
(подпись)

Ставрополь 2022

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

Порядок выполнения работы:

1. Создал общедоступный репозиторий на GitHub, в котором использована лицензия MIT и язык программирования Python.

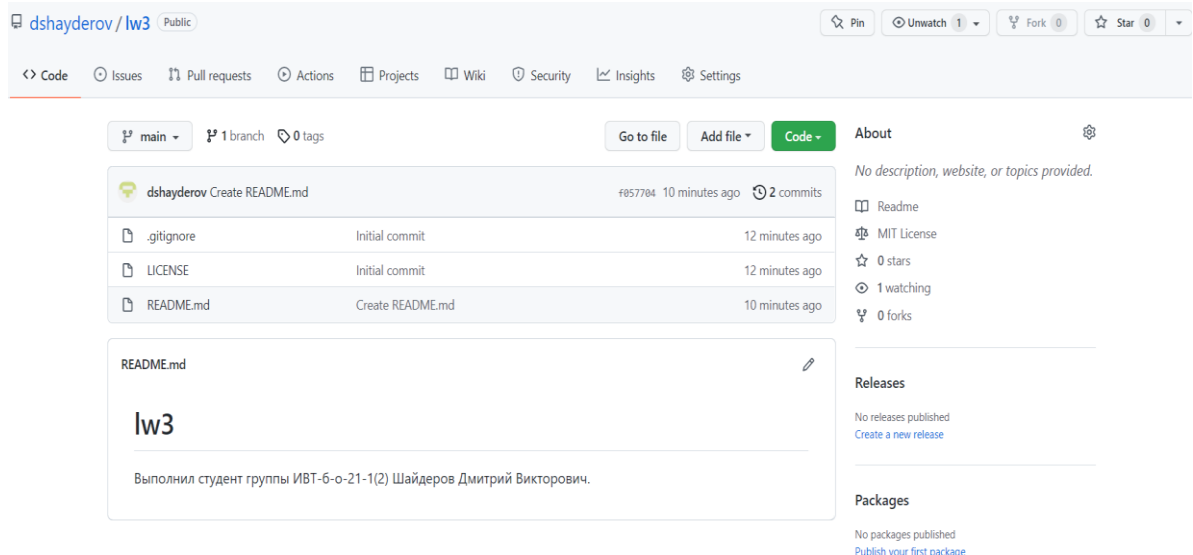


Рисунок 1 - Созданный репозиторий

2. Выполнил клонирование созданного репозитория.

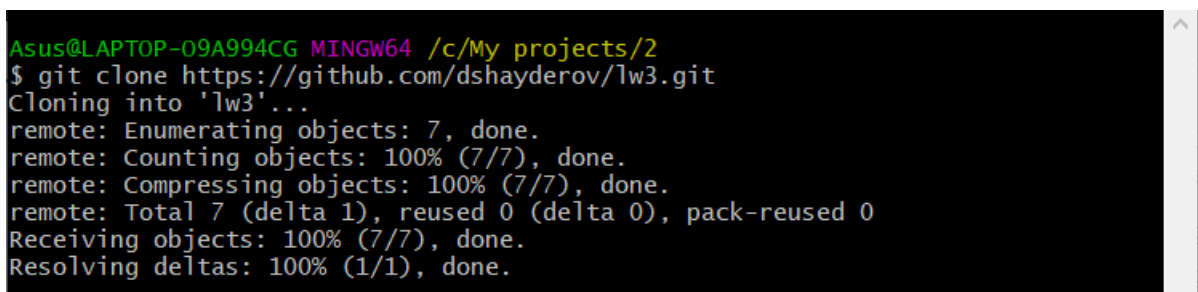


Рисунок 2 - Клонирование репозитория

3. Дополнил файл .gitignore необходимыми правилами для работы с IDE PyCharm.

```
122
123 # .pyru
124 .pyru_cache/
125 .dmyru.json
126 dmyru.json
127
128 # Pyre type checker
129 .pyre/
130
131 PyCharm делает разработку максимально продуктивной благодаря функции автодополнения и анализа кода, мгновенной подсветке ошибок и быстрым исправлениям. Автоматические рефакт
132 Умный редактор PyCharm предназначен для максимально продуктивной разработки на Python, JavaScript, CoffeeScript, TypeScript, CSS и популярных языках шаблонов. Функции автодоп
133 Умный поиск позволяет быстро перейти к любому классу, файлу или символу, а также к нужному окну или действию IDE. Переход к выделенному методу, тесту, объявлению, вхождению
134 PyCharm предоставляет широкие возможности реорганизации кода с помощью рефакторингов Rename и Delete, Extract Method, Introduce Variable, Inline Variable, Inline Method и мно
135 PyCharm предлагает большой набор инструментов из коробки: встроенный отладчик и инструмент запуска тестов, профилировщик Python, полнофункциональный встроенный терминал, инст
136 С PyCharm вы сможете работать с ноутбуками Jupyter, запускать команды в интерактивной консоли Python, подключать библиотеки Anaconda, а также работать с другими библиотеками
137 PyCharm можно установить на Windows, macOS и Linux с помощью одного лицензионного ключа. Настройте рабочую среду так, как вам нравится: выберите подходящую цветовую схему и у
```

Рисунок 3 - Файл .gitignore

4. Создал проект PyCharm в папке репозитория.

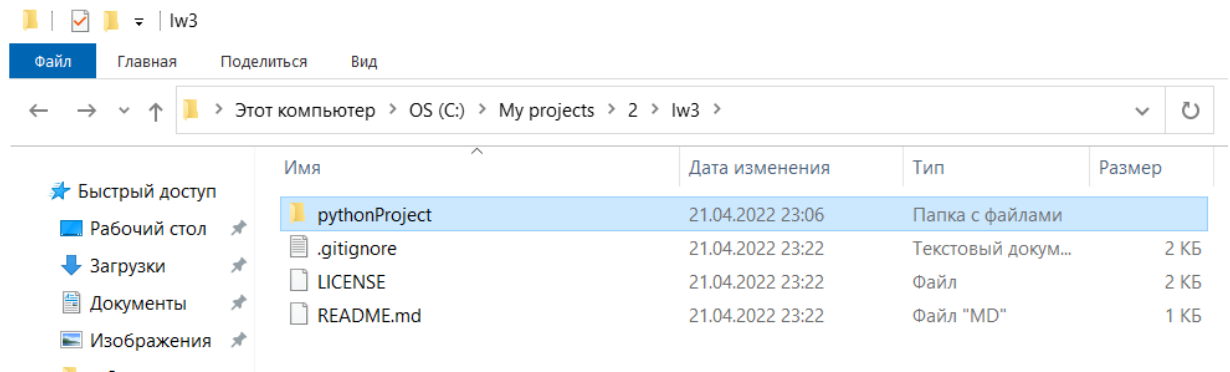


Рисунок 4 - Создание проекта

5. 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?")

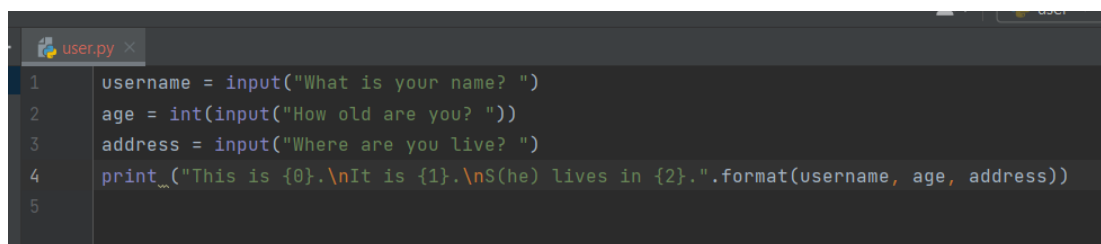


Рисунок 5 - Код программы

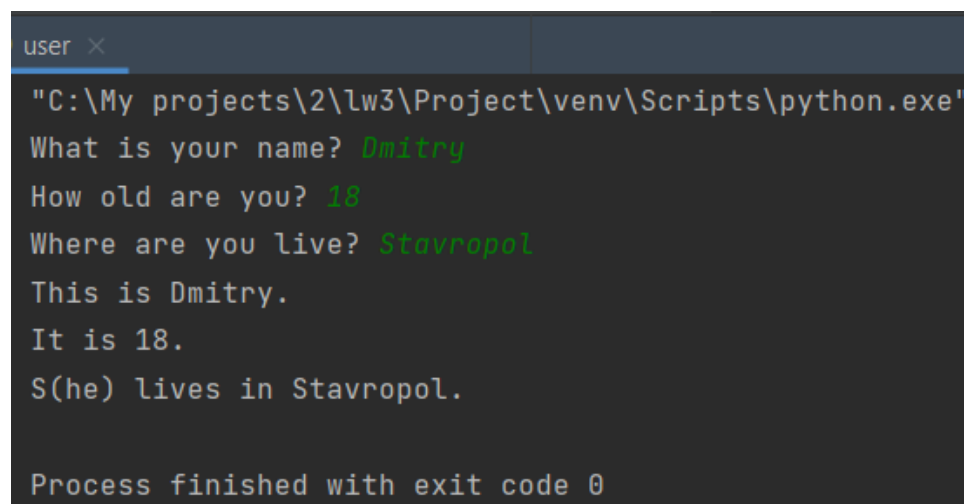
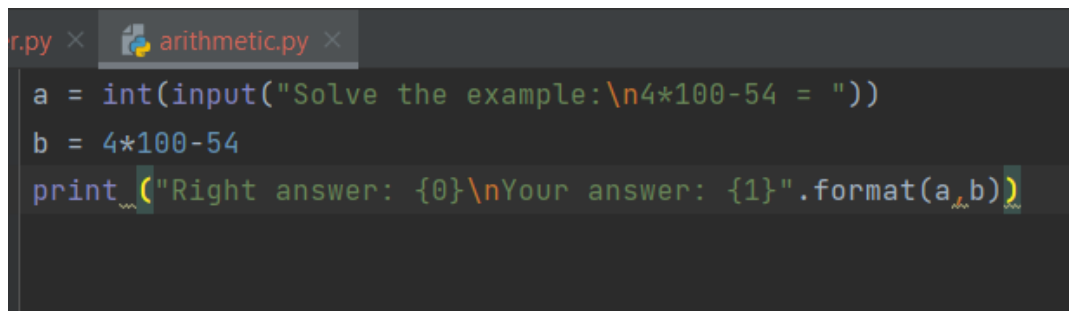


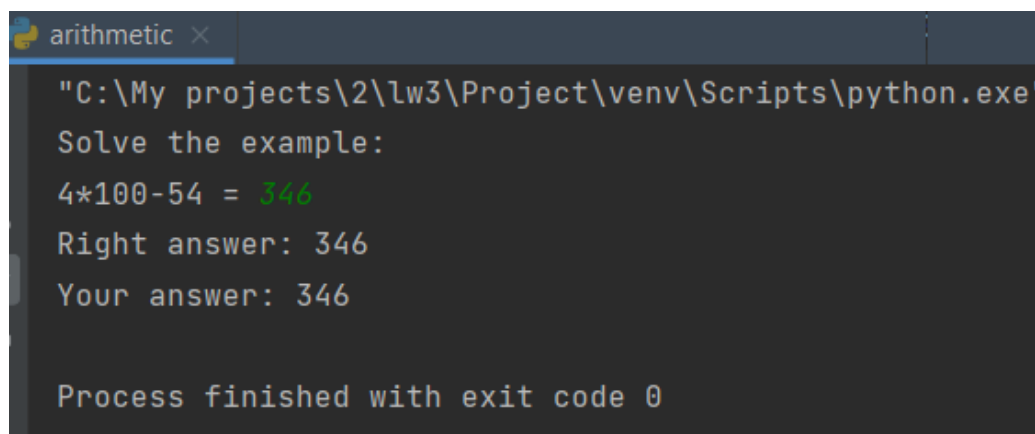
Рисунок 6 - Результат выполнения программы

6. Написал программу (файл arithmetic.py), которая предлагает пользователю решить пример $4 * 100 - 54$, потом выводит на экран правильный ответ и ответ пользователя.



```
arithmetic.py x
a = int(input("Solve the example:\n4*100-54 = "))
b = 4*100-54
print("Right answer: {0}\nYour answer: {1}".format(a,b))
```

Рисунок 7 - Код программы

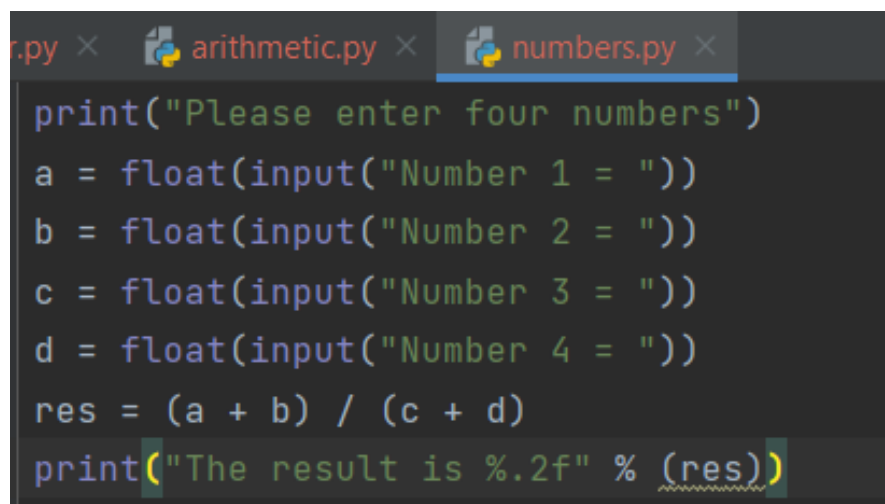


```
arithmetic x
"C:\My projects\2\lw3\Project\venv\Scripts\python.exe
Solve the example:
4*100-54 = 346
Right answer: 346
Your answer: 346

Process finished with exit code 0
```

Рисунок 8 - Результат выполнения программы

7. Программа запрашивает у пользователя четыре числа (файл numbers.py), отдельно складывает первые два и отдельно вторые два, делит первую сумму на вторую и выводит результат на экран так, чтобы ответ содержал две цифры после запятой.



```
numbers.py x
print("Please enter four numbers")
a = float(input("Number 1 = "))
b = float(input("Number 2 = "))
c = float(input("Number 3 = "))
d = float(input("Number 4 = "))
res = (a + b) / (c + d)
print("The result is %.2f" % (res))
```

Рисунок 9 - Код программы

```
numbers ×
"C:\My projects\2\lw3\Project\venv\Scripts\python.exe"
Please enter four numbers
Number 1 = 21.1
Number 2 = 33.3
Number 3 = 12.5
Number 4 = 23.7
The result is 1.50

Process finished with exit code 0
```

Рисунок 10 - Результат выполнения программы

8. Написал программу (файл individual.py) для решения индивидуального задания.

Вариант 3: Даны стороны треугольника , и найти площадь треугольника по формуле Герона.

```
individual.py × arithmetic.py × numbers.py × individual.py ×
import math
print("Area of triangle")
a = float(input("a = "))
b = float(input("b = "))
c = float(input("c = "))
p = (a+b+c)/2
s = float(math.sqrt(p*(p-a)*(p-b)*(p-c)))
print("s = %.2f" % (s))
```

Рисунок 11 - Код программы

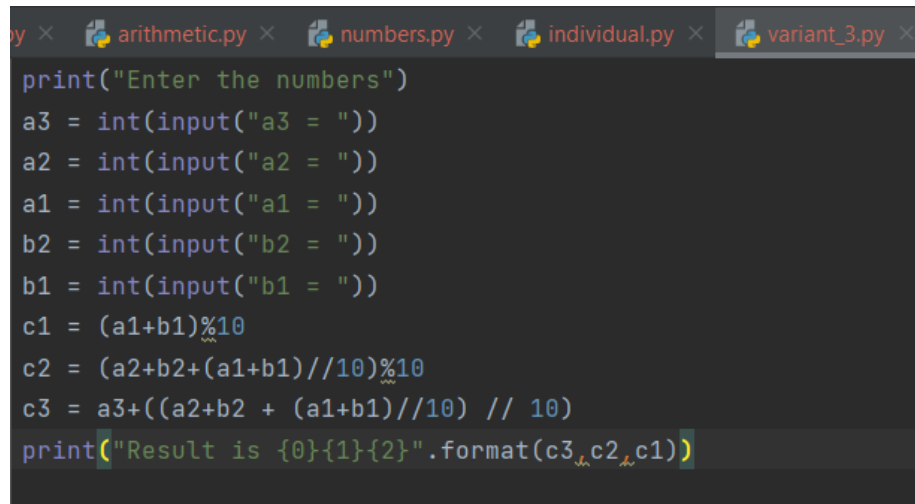
```
individual ×
"C:\My projects\2\lw3\Project\venv\Scripts\python.exe"
Area of triangle
a = 3.6
b = 4.1
c = 5.2
s = 7.35

Process finished with exit code 0
```

Рисунок 12 - Результат выполнения программы

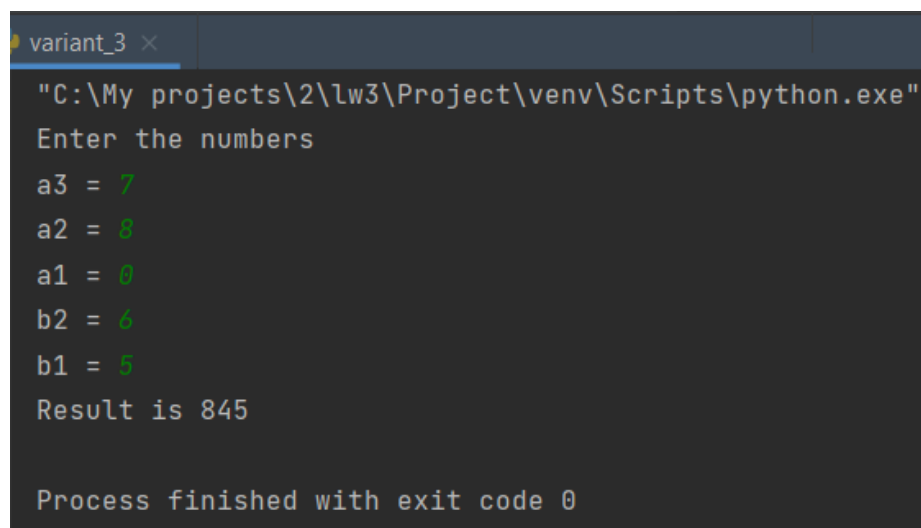
9. Написал программу (файл variant_3.py) для решения задачи повышенной сложности.

Вариант 3: Даны цифры двух десятичных целых чисел: трехзначного и двузначного, где a_3 – число единиц, a_2 – число десятков, a_1 – число сотен. Получить цифры числа, равного сумме заданных чисел (известно, что это число трехзначное). Числа-слагаемые и число-результат не определять; условный оператор не использовать.



```
print("Enter the numbers")
a3 = int(input("a3 = "))
a2 = int(input("a2 = "))
a1 = int(input("a1 = "))
b2 = int(input("b2 = "))
b1 = int(input("b1 = "))
c1 = (a1+b1)%10
c2 = (a2+b2+(a1+b1)//10)%10
c3 = a3+((a2+b2 + (a1+b1)//10) // 10)
print("Result is {0}{1}{2}".format(c3,c2,c1))
```

Рисунок 13 - Код программы



```
variant_3 x
"C:\My projects\2\lw3\Project\venv\Scripts\python.exe"
Enter the numbers
a3 = 7
a2 = 8
a1 = 0
b2 = 6
b1 = 5
Result is 845

Process finished with exit code 0
```

Рисунок 14 - Результат выполнения программы

10. Выполнил коммит и пуш файлов user.py, arithmetic.py, numbers.py и individual.py в репозиторий

```

Asus@LAPTOP-09A994CG MINGW64 /c/My projects/2/lw3 (main)
$ git pull
Already up to date.

Asus@LAPTOP-09A994CG MINGW64 /c/My projects/2/lw3 (main)
$ git add .
warning: LF will be replaced by CRLF in Project/.idea/inspectionProfiles/Project_Default.xml.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in Project/.idea/inspectionProfiles/profiles_settings.xml.
The file will have its original line endings in your working directory

Asus@LAPTOP-09A994CG MINGW64 /c/My projects/2/lw3 (main)
$ git commit -m "Выполнение работы"
[main 261ee11] Выполнение работы
12 files changed, 75 insertions(+)
create mode 100644 Project/.idea/.gitignore
create mode 100644 Project/.idea/Project.iml
create mode 100644 Project/.idea/inspectionProfiles/Project_Default.xml
create mode 100644 Project/.idea/inspectionProfiles/profiles_settings.xml
create mode 100644 Project/.idea/misc.xml
create mode 100644 Project/.idea/modules.xml
create mode 100644 Project/.idea/vcs.xml
create mode 100644 Project/arithmetic.py
create mode 100644 Project/individual.py
create mode 100644 Project/numbers.py
create mode 100644 Project/user.py
create mode 100644 Project/variant_3.py

Asus@LAPTOP-09A994CG MINGW64 /c/My projects/2/lw3 (main)
$ git push
Enumerating objects: 18, done.
Counting objects: 100% (18/18), done.
Delta compression using up to 8 threads
Compressing objects: 100% (16/16), done.
Writing objects: 100% (17/17), 2.58 KiB | 1.29 MiB/s, done.
Total 17 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/dshayderov/lw3.git
   a57a9e3..261ee11  main -> main

Asus@LAPTOP-09A994CG MINGW64 /c/My projects/2/lw3 (main)
$ |

```

Рисунок 15 - Коммит и пуш изменений

Ответы на контрольные вопросы:

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

Чаще всего интерпретатор Python уже входит в состав дистрибутива.

Windows: Осн. этапы установки Python на Windows:

1. Скачать дистрибутив с официального сайта;
2. Запустить скачанный установочный файл;
3. Выбрать способ установки;
4. Отметить необходимые опции установки;
5. Выбрать место установки;
6. Готово.

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

Пакет Anaconda содержит версии языка Python 2 и 3, набор наиболее часто используемых библиотек и удобную среду разработки и исполнения, запускаемую в браузере, а также на Anaconda удобнее запускать примеры.

3. Как осуществить проверку работоспособности пакета Anaconda?

Для выполнения проверки работоспособности Anaconda необходимо вначале запустить командный процессор с поддержкой виртуальных окружений Anaconda. В появившейся командной строке необходимо ввести `> jupyter notebook`, в результате чего отобразится процесс загрузки веб-среды Jupyter Notebook, после чего запустится веб-сервер и среда разработки в браузере. Создать ноутбук для разработки, для этого нажать на кнопку New и в появившемся списке выбрать Python. В результате будет создана новая страница в браузере с ноутбуком. Ввести в первой ячейке команду `print("Hello, World!")` и нажать Alt+Enter на компьютере. Ниже ячейки должна появиться соответствующая надпись.

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

Указать путь до интерпретатора в настройках IDE, для этого:

- 1) нажмите на шестеренку в верхнем правом углу, выберите "Add..";
- 2) далее выберите "System Interpreter";
- 3) нажмите на 3 точки "..." справа от поля в выборе интерпретатора;
- 4) укажите путь до интерпретатора.

5. Как осуществить запуск программы с помощью IDE PyCharm?

Сочетанием клавиш Shift+F10.

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

Интерактивный. Python можно использовать как калькулятор для различных вычислений, а если дополнительно подключить необходимые математические библиотеки, то по своим возможностям он становится

практически равным таким пакетам как Matlab, Octave и т.п. Пакетный. В этом режиме сначала записывается вся программа, а потом эта программа выполняется полностью.

7. Почему язык программирования Python называется языком динамической типизации?

Т. к. в ЯП Python проверка типа происходит во время выполнения, а не компиляции.

8. Какие существуют основные типы в языке программирования Python?

Типы в ЯП Python:

1. None.
2. Логические переменные.
3. Числа.
4. Списки.
5. Строки.
6. Бинарные списки.
7. Множества.
8. Словари.

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. При инициализации переменной, на уровне интерпретатора, создается целочисленный объект, который имеет некоторый идентификатор, значение и тип. Посредством оператора “=” создается ссылка между переменной и объектом.

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()`?

Функция `id()` предназначена для получения значения идентичности объекта. С помощью функции `type()` можно получить тип конкретного объекта.

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`).

13. Чем отличаются операции деления и целочисленного деления?

При целочисленном делении отбрасывается дробная часть от деления чисел, при операции деления дробная часть не отбрасывается.

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 (`x.real`) и мнимую части (`x.imag`). Для получения комплексносопряженного числа необходимо использовать метод `conjugate()`.

15. Каково назначение и основные функции библиотеки (модуля) `math`? По аналогии с модулем `math` изучите самостоятельно назначение и основные функции модуля `cmath`.

Для выполнения математических операций необходим модуль `math`.

Осн. операции библиотеки math:

math.ceil(x) - возвращает ближайшее целое число большее, чем x.

math.fabs(x) - возвращает абсолютное значение числа.

math.factorial(x) - вычисляет факториал x.

math.floor(x) - возвращает ближайшее целое число меньшее, чем x.

math.exp(x) - вычисляет e^x . math.log2(x) - логарифм по основанию 2.

math.log10(x) - логарифм по основанию 10.

math.log(x[, base]) - по умолчанию вычисляет логарифм по основанию e, дополнительно можно указать основание логарифма.

math.pow(x, y) - вычисляет значение x в степени y.

math.sqrt(x) - корень квадратный от x.

math.cos(x) - косинус от x.

math.sin(x) - синус от x.

math.tan(x) - тангенс от x.

math.acos(x) - арккосинус от x.

math.asin(x) - арксинус от x.

math.atan(x) - арктангенс от x.

math.pi - число пи.

math.e - число e.

16. Каково назначение именных 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.

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.

Форматирование может выполняться в так называемом старом стиле или с помощью строкового метода format. Символы %s , %d , %f

подставляются значения переменных. Буквы s, d, f обозначают типы данных – строку, целое число, вещественное число.

Начиная с версии 3.6 в Python появился новый тип строк — **f-строки**, которые буквально означают «formatted string». Эти строки улучшают читаемость кода, а также работают быстрее чем другие способы форматирования. F-строки задаются с помощью литерала «f» перед кавычками.

f-строки делают очень простую вещь — они берут значения переменных, которые есть в текущей области видимости, и подставляют их в строку. В самой строке вам лишь нужно указать имя этой переменной в фигурных скобках.

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

Указать перед input тип данных: int(input()) или float(input()).

Вывод: исследовал процесс установки и базовые возможности языка Python версии 3.x.