

# Python. Знакомство. Типы данных, вычисление и переменные.

# Знакомство с языком программирования Python

- Компьютерная программа - набор инструкций, следуя которым компьютер выполняет различные действия. Практически любое из электронных устройств, которыми мы пользуемся, не будет работать или станет гораздо менее полезным, если лишить его программного обеспечения.
- Как и люди, компьютеры «говорят» на разных языках, только языки эти — компьютерные. Компьютерный язык служит для того, чтобы переговариваться с компьютером, используя команды, понятные и компьютеру, и человеку.

# Установка.

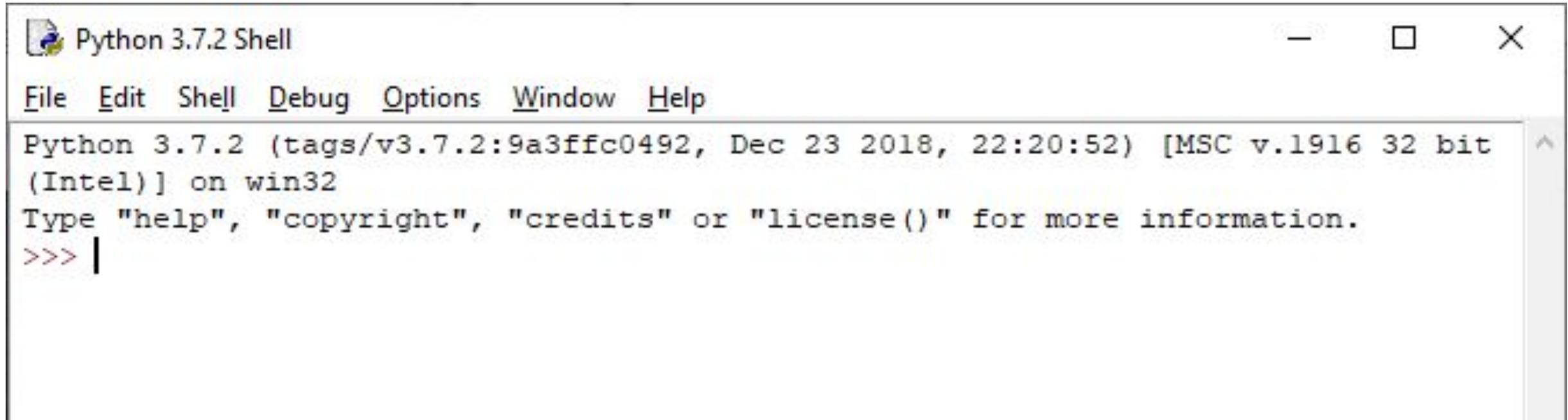
The screenshot shows the Python.org website with the navigation menu open. The 'Downloads' menu item is selected, showing a list of options: All releases, Source code, Windows, Mac OS X, Other Platforms, License, and Alternative Implementations. The 'Windows' option is highlighted, and a sub-menu is displayed with the following content:

- Download for Windows**
- Python 3.7.4
- Note that Python 3.5+ cannot be used on Windows XP or earlier.**
- Not the OS you are looking for? Python can be used on many operating systems and environments.
- [View the full list of downloads.](#)

At the bottom of the page, there is a blue banner with the text: "Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)"



- Это командная оболочка Python, которая входит в интегрированную среду разработки, а три знака «больше» (>>>) называются приглашением. После приглашения можно вводить различные команды.



```
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> |
```

# Python и простые математические ВЫЧИСЛЕНИЯ

- Если нужно перемножить два числа, к примеру узнать, сколько будет  $8 \times 3,57$ , мы обычно пользуемся калькулятором либо берем ручку и умножаем в столбик на листе бумаги. А что если использовать для подсчетов оболочку Python? После значка `>>>` введите выражение и нажмите Enter.

```
>>> 2019-1990  
29
```

- При записи числа 3,57 используется не запятая, а точка. Кроме того, в Python числа перемножаются с помощью звездочки (\*), а не знака умножения (×).
- В оболочке Python можно умножать, складывать, вычитать и делить числа, а также совершать некоторые другие операции. Символы, с помощью которых выполняются математические действия в языке Python, называются операторами

Символ	Операция
+	Сложение
-	Вычитание
*	Умножение
/	Деление

# Что такое переменная

- Но что делать если необходимо высчитать значение, а затем использовать его результат дальше? На помощь к нам приходят переменные.
- В программировании слово переменная обозначает именованное место для хранения данных, например, чисел, текста, списков с числами или символами и так далее.
- Имя переменной стоит задавать так, чтобы было понятно, что же она хранит.

- Например, мы хотим узнать возраст, а затем этот возраст вывести на экран с дополнительным текстом. Для этого, создадим переменную `age` и в нее сохраним возраст. Для этого мы пишем имя переменной, ставим знак `=` и пишем наше выражение. Знак `=` это оператор присваивания. т.е. мы в переменную записываем наше значение.

```
>>> age = 2019-1990  
>>> |
```

- И мы видим, что результат у нас не вывелся на экран, как было в прошлый раз. Теперь наш результат хранится в переменной, и мы его можем использовать повторно, без необходимости вычисления выражения заново.
- Чтобы увидеть, что же хранится в нашей переменной мы можем просто написать ее имя.

```
>>> age = 2019-1990
>>> age
29
>>> |
```

- Если мы хотим вывести на экран вместе с значением дополнительный текст мы можем воспользоваться специальной функцией print().

```
>>> print("Hello")
Hello
>>> print(2+3)
5
>>> print(age)
29
>>> print("Возраст человека", age, "лет")
Возраст человека 29 лет
>>> |
```

# Практика

- 1.1 Посчитать и вывести на экран возраст членов семьи, чтобы вывело на экран текст в следующем виде. Значения хранить в переменных: Папе 55 лет

## Решени

```
>>> mum = 2019-1965
>>> papa = 2019-1964
>>> sister = 2019-1990
>>> me = 2019-1991
>>> print("Маме", mum, "лет")
Маме 54 лет
>>> print("Папе", papa, "лет")
Папе 55 лет
>>> print("Сестре", sister, "лет")
Сестре 29 лет
>>> print("Мне", me, "лет")
Мне 28 лет
>>> |
```



# Типы данных

- **Тип данных** — допустимое множество значений. Зная тип данных мы также знаем какие действия мы можем с ним делать.
- Для того чтобы узнать какой тип данных имеет та или иная переменная, в Python есть функция `type()`.

```
>>> type(23)
```

```
<class 'int'>
```

```
>>> type(23.5)
```

```
<class 'float'>
```

```
>>> |
```

Целые числа - тип int.

Дробные числа - тип float.

```
>>> type(age)
```

```
<class 'int'>
```

```
>>> |
```



- В функции `print()` мы выводили текст. Этот текст мы писали в кавычках.
- Текст в Python это строковый тип данных. Проверим.

```
>>> type("text")  
<class 'str'>  
>>> |
```

Строка представляет собой список отдельных букв, символов. Представьте себе книгу. У нее есть список страниц. И когда вы читаете одну из страниц, вы как бы обращаетесь к отдельному элементу из общего списка, по ее индексу – номеру страницы. Так и со строками. Мы можем получить ее отдельный элемент по индексу.



# Допустимые операции.

- К числовым типам, данных применимы все операции что и в обычной жизни. Сложение, вычитание, умножение, деление, возведение в степень, извлечение корня и т.д.
- К строкам применимы операции сложения и умножения. Операция сложения позволяет склеить две строки в одну. А операция умножения на число позволяет продублировать строки несколько раз

```
>>> "Hello "+"World"
'Hello World'
>>> "Hello"*3
'HelloHelloHello'
>>> |
```



# Input()

- Сейчас мы умеем работать в программе, в которой все данные задаются внутри. Но ведь во многих программах мы можем вводить свои данные. Имя, например. В питоне мы также можем это сделать. В этом нам поможет команда `input()`. Это также стандартная функция Python. Однако стоит помнить, что все что мы вводим с клавиатуры, считается текстовым типом данных, даже если мы ввели число.

```
>>> name = input()  
Petya  
>>> name  
'Petya'  
>>> |
```



- Если мы хотим вывести при этом дополнительную информацию мы можем передать текст внутри скобок.

```
>>> name = input("Введите ваше имя: ")  
Введите ваше имя: Petya
```

# Практика

- Необходимо обменять значение двух переменных.  $a = 5$ ,  $b = 10$
- На выходе должно в переменной  $a$  храниться значение 10, в переменной  $b$  значение 5. Просто присвоить новые значения в переменные нельзя! Необходимо произвести именно обмен значениями.

# Практика

- Теперь программа предлагает нам ввести имена и возраст членов семья. а затем выводит имя и возраст. Программу пишем в файле.

Мою маму зовут: Алла

Ее возраст: 54

Мама Алла ей 54 года

## Решени

```
name = input()
age = input()
print("Мою маму зовут: ", name)
print("Её возраст: ", age)
print("Мама ", name, " ей ", age, " года")
```



# ДЗ

- 1. У вас заданы стороны прямоугольника. Рассчитать и вывести на экран его периметр.
- 2. С клавиатуры вводятся две стороны прямоугольника. Рассчитать его периметр и площадь. Вывести на экран.
- 3. Посчитать сумму цифр трехзначного числа. Для выполнения данной задачи необходимо познакомиться с операцией нахождения целого при делении нацело (“//”) и операцией нахождения остатка при делении нацело (“%”).