

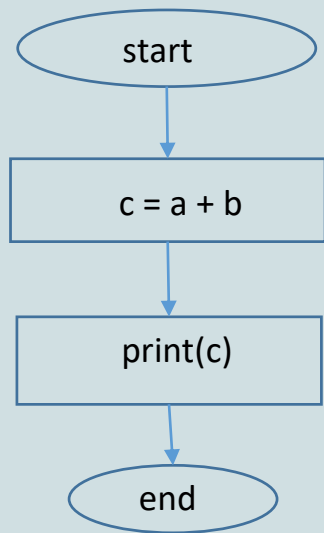
Вебинар №5

Python, базовый курс

Алгоритмы

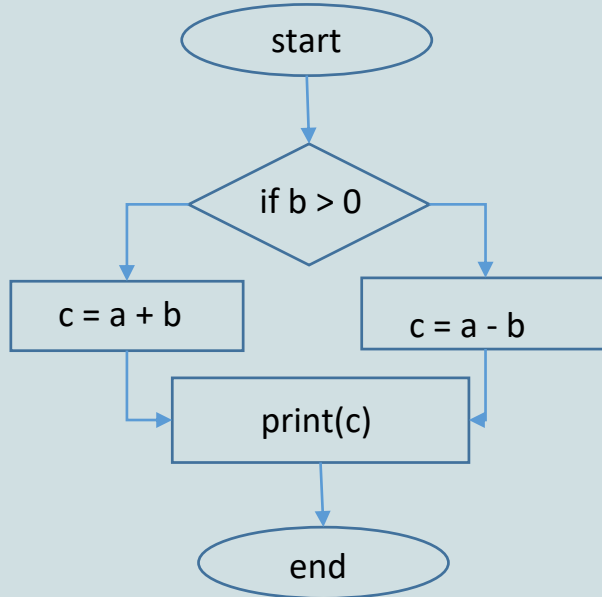
Линейные

команды выполняются
последовательно



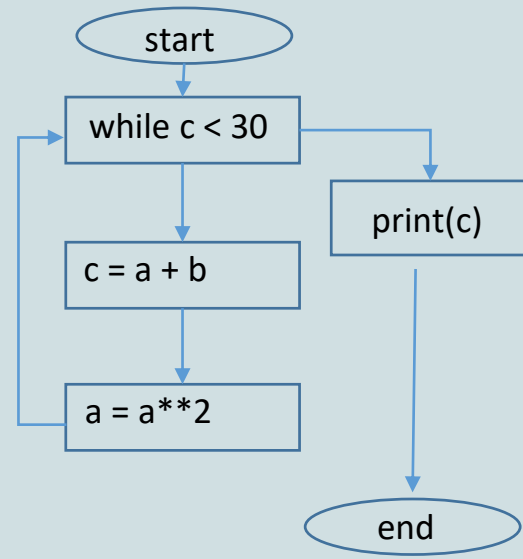
С ветвлением

алгоритм имеет несколько веток решения
в зависимости от значений параметров

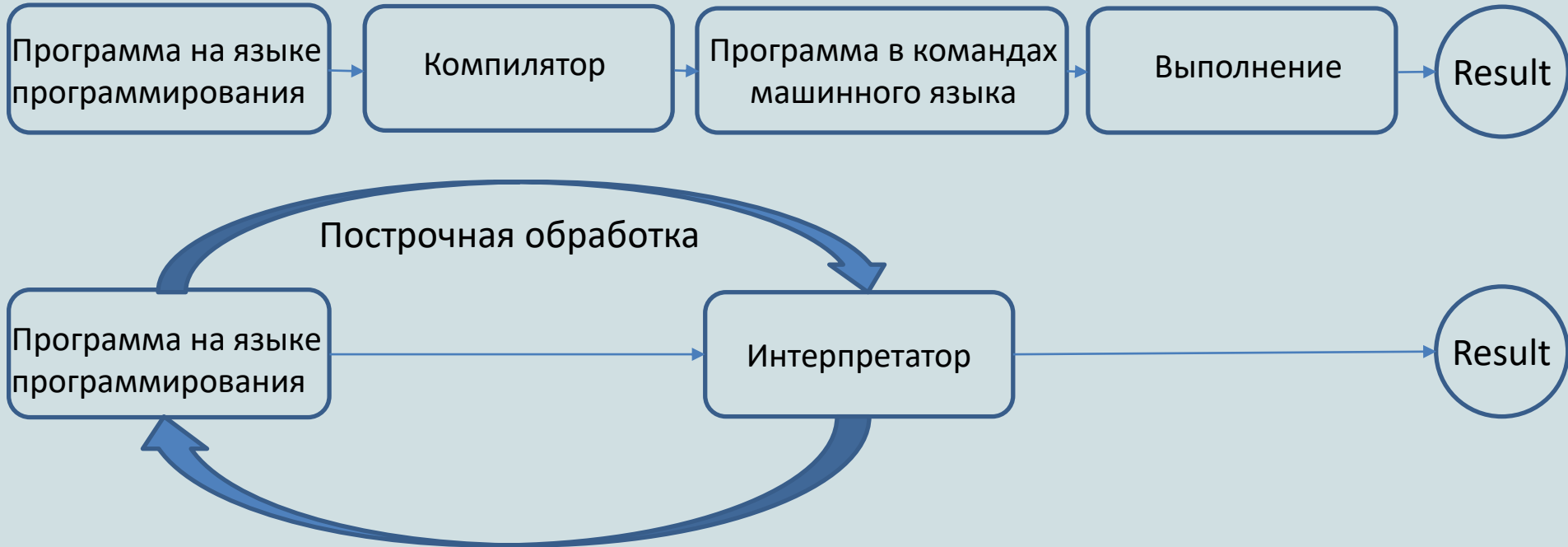


Циклические

команды повторяются до конца
цикла



Интерпретируемые и компилируемые языки программирования





Python

Python — интерпретируемый язык программирования.

Для выполнения языка python написан интерпретатор (CPython, JPython, ...), который выполняет программу **построчно**. Интерпретатор это программа, написанная на другом языке программирования, которая понимает конструкции языка python. Например, понимает все логические конструкции, осуществляет хранение объектов используемых в программе во время выполнения и т.д.

Как запускать программы на python?

Для выполнения программ на своём компьютере, нужно установить интерпретатор.

<https://www.python.org/downloads/>





Как запускать программы на python?

Для выполнения программ на своём компьютере, нужно установить интерпретатор.

Но выполнять файлы по отдельности не очень удобно, и в реальности аналитики чаще всего работают в интерактивной среде [jupyter notebook](#).

В этой среде, похожей на блокнот, интерпретатор уже запущен, и вы можете сразу выполнять код в ячейках. При этом, поскольку по факту выполняется одна программа, переменные и функции из одной ячейки будут доступны в других ячейках после того как первая ячейка была выполнена.

Также для разработки могут быть использованы PyCharm и Visual Studio Code

Google Colaboratory

The screenshot shows the Google Colaboratory web interface. At the top, the 'master.ipynb' file is open. The menu bar includes 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. A status bar at the top right shows 'RAM', 'Disk', and 'Editing' status. The main content area displays a Jupyter Notebook with the following text:

Занятие 1

Примитивы

Числа

добавить ячейку с текстом

добавить ячейку с кодом

float — числа с плавающей точкой

Чтобы посмотреть тип данных существует встроенная функция `type()`

```
[1] type(3)
```

```
int
```

```
[2] type(3.14)
```

```
float
```

В примере ниже `pi` — название переменной (`name`), а `3.14` это её значение (`value`)

```
[3] pi = 3.14
```

```
type(pi)
```

```
float
```

Для чисел также доступны базовые арифметические операции:

```
[ ]
```

Строки

Строки используются достаточно часто.

Строки — последовательности символов, любых, которые вы можете набрать на клавиатуре.

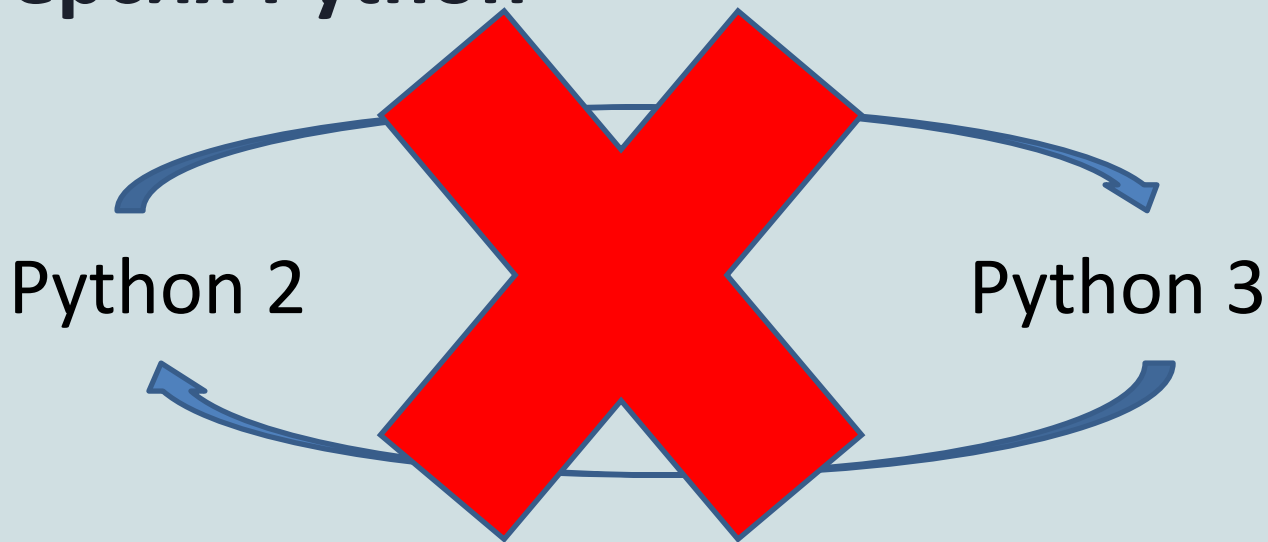
В Python строки обозначаются кавычками двойными или одинарными, без разницы

Annotations on the image:

- А black arrow points from the text 'следите за тем, чтобы ваши изменения сохранялись' to the 'Help' menu item.
- Two black arrows point from the text 'добавить ячейку с текстом' to the '+ Text' button in the top left.
- A black arrow points from the text 'добавить ячейку с кодом' to the '+ Code' button in the top left.
- A black arrow points from the text 'ячейка с кодом' to the first code cell containing `type(3)`.
- A black arrow points from the text 'ячейка с текстом' to the second code cell containing `int`.
- A black arrow points from the text 'кнопка "поделиться"' to the 'Share' button in the top right.



Версия Python



Узнать версию на ПК, ввести в cmd: `python`

Узнать версию в Colab: `!python -v`



PEP8 и философия Python

«Код читается намного больше раз, чем пишется»

- Явное лучше, чем неявное.
 - Простое лучше, чем сложное.
 - Сложное лучше, чем запутанное.
 - Плоское лучше, чем вложенное.
 - Читаемость имеет значение.
 - Ошибки никогда не должны замалчиваться.
 - Если они не замалчиваются явно.
 - Встретив двусмысленность, отбрось искушение угадать.
 - Должен существовать один и, желательно, только один очевидный способ сделать это.
 - Если реализацию сложно объяснить, то это плохая идея.
 - Если реализацию легко объяснить, то идея, возможно, хороша.
-



Онлайн ввод и вывод, переменные



input() и print()

- Когда интерпретатор встречает в коде вызов `input()`, он приостанавливает выполнение и ожидает пользовательского ввода из консоли.
- Когда интерпретатор встречает в коде вызов `print()`, он выводит в консоль то, что было передано при вызове `print()`.

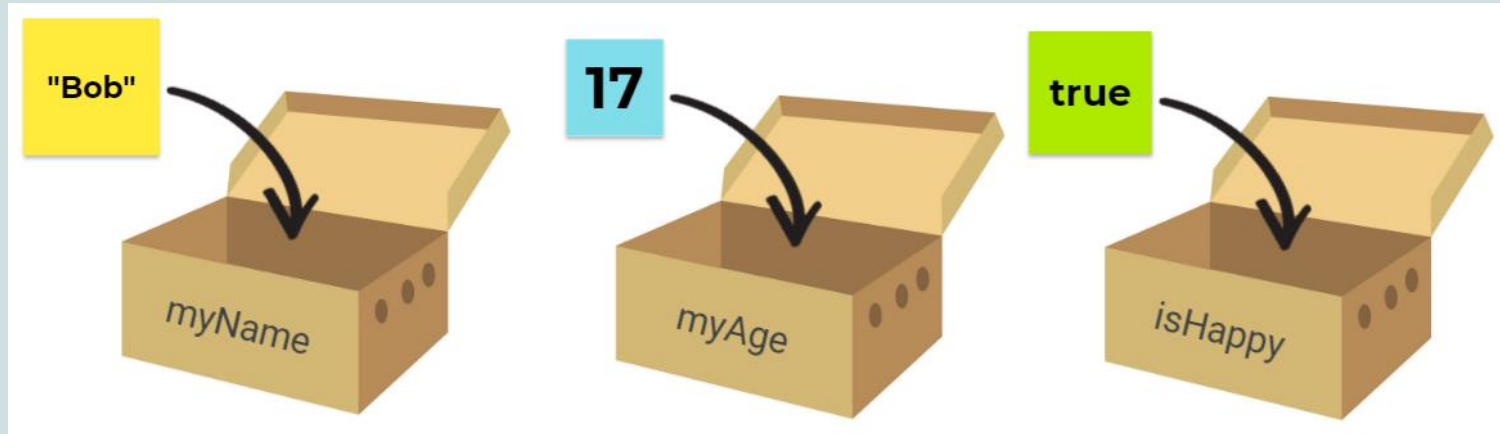
Например, простая программа

```
print("Hello world")
```

выведет на экран Hello world, и закончит выполнение.

Базовые понятия. Переменная

Переменная в императивном программировании — поименованная, либо адресуемая иным способом область памяти, адрес которой можно использовать для осуществления доступа к данным. Данные, находящиеся в переменной (то есть по данному адресу памяти), называются значением этой переменной.





Обработка ввода

Как правило, программы сложнее, чем просто ввод и вывод. Они обрабатывают ввод, чтобы выдать какой-то ожидаемый результат. В частности, ввод может быть из консоли.

Чтобы положить ввод в переменную в языке python, нужно объявить переменную.

```
my_variable = input()
```

Кроме того, мы можем задать значение, на которое будет ссылаться переменная сразу

```
variable2 = 1
```

```
variable3 = 3.14
```

А также **присваивать** значение в одной переменной, на значение в другой.

```
variable2 = variable3
```

```
|
```



Задачи

Задача 1

На вход даны переменные `a`, `b`

Поменяйте значения, на которые ссылаются эти переменные, местами

Задача 2

Запишите ввод из консоли в переменную `my_first_variable` и выведите ее на экран с помощью `print()`

Задача 3

Примите на вход пользовательский ввод два раза, и сначала выведите на экран второй ввод, а затем первый

Задача 4

Еще немного про `print()`.

`print()` может выводить сразу несколько значений, при этом можно заранее указать с каким разделителем нужно вывести эти значения



Встроенные типы данных в python



Что значит встроенные?

Это означает, что не нужно подключать какие-то сторонние библиотеки, чтобы с ними работать. Также, это означает, что эти типы идут с определенными свойствами и доступным операциями “из коробки”.

Можно также сказать, что из них “состоят” более сложные, пользовательские абстракции.



Встроенные типы

- “Простые”
 - Numbers (числа)
 - Целые числа (int)
 - Числа с плавающей точкой (float)
 - Комплексные числа
 - Strings (строки) (str)
 - Boolean (логический тип данных) (bool)
- Коллекции
 - Lists (списки)
 - Dictionaries (словари)
 - Tuples (кортежи)
 - Sets (множества)



Действия с простыми типами данных:

Посмотреть тип переменной – функция `type()`

Числа (целые и нецелые) можно:

- складывать (+),

- вычитать (-),

- умножать (*),

- делить (/)

- возводить в степень (**),

- находить остаток от деления(%)

- находить целую часть при делении (//)

Строки можно соединять между собой (конкатенировать) `"abc"+"def" → "abcdef"`

Кроме того, мы можем переходить от одного типа к другому с помощью встроенных в язык функций:

- `int()`

- `float()`

- `str()`

- `bool()`

Например, преобразовать число с плавающей точкой в целое: `int(3.2) → 3`



Задача 5

На вход программе подаётся два целых числа, каждое на отдельной строке. Вычислите их сумму

Формат входных данных

На вход программе подаются два числа, каждое на отдельной строке.

Формат выходных данных

Программа должна вывести одно число, равное сумме введенных

Задача 6

На вход подается одно целое число. Выведите три числа идущих за ним

Формат входных данных

На вход программе подаются одно целое число

Формат выходных данных

Программа должна вывести три следующих числа

Задача 7

Дано два числа. Выведите длину гипотенузы треугольника с заданными катетами.

Формат входных данных

На вход программе подаются два числа, каждое на отдельной строке.

Формат выходных данных

Программа должна вывести одно число, равное длине гипотенузы



Задача 8

Какое из представленных ниже значений можно привести к типу float за одну операцию

- A) 'd543'
- B) '23.4'
- C) -12.12
- D) 562 + ' 5'
- E) '562 + 5'
- F) 'e5V7'

Задача 9

Используя стандартные математические операции получить наибольшее значение из чисел 2, 3, 3

Задача 10

Дана площадь сферы S , найдите ее радиус.

P.S.: площадь сферы вычисляется по формуле $S = 2 \cdot \pi \cdot R^2$



Управляющие конструкции: if (elif) else



Зачем?

Для того, чтобы проверять выполнение различных условий

Общий скелет конструкции

- If-else

```
if a < b:  
    print "Computer says no"  
else:  
    print "Computer says yes"
```

- If-elif-else

```
if a < b:  
    print "Computer says not enough"  
elif a > b:  
    print "Computer says too much"  
else:  
    print "Computer says just right"
```



Задача 11

Подается на вход целое число n . Определить делится ли число на 3, если да, вывести на экран результат деления на 3, а если нет, то вывести остаток от деления на 3 на экран.

Задача 12

Напишите программу, которая отвечает YES, если точка находится в I или III четверти и NO иначе. На вход подаются две координаты, сначала x , а затем y (по отдельности, каждая в новом вводе)

Задача 13

Реализуйте на python алгоритм игры 'холодно-горячо' для чисел от 1 до 20. Правильный ответ 9

Задача 14

Напишите программу, которая преобразует любое число в число с плавающей точкой. Если в качестве аргументы передана строка, то необходимо возвращать 'Перевод невозможен'



Обратная связь

https://docs.google.com/forms/d/e/1FAIpQLScLLI5Vno4ynQVLn4Tk5CifvSxYueYxFXcWQfybxQ1UvPSoXQ/viewform?usp=sf_link