

# Python 3

Python3 – высокоуровневый язык программирования общего назначения, с помощью которого можно создавать высокопроизводительный и понятный код. Стандартная библиотека языка включает в себя большой объем полезного функционала. Python 3 поддерживает структурное, объектно-ориентированное, функциональное, императивное и аспектно-ориентированное программирование. Основные архитектурные черты – динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений, высокоуровневые структуры данных. Поддерживается разбиение программ на модули, которые, в свою очередь, могут объединяться в пакеты.

Язык поддерживает основные понятия объектно-ориентированного программирования (классы; наследование, в том числе множественное; полиморфизм (все функции виртуальные); инкапсуляция; конструкторы, деструкторы, распределители памяти; перегрузка операторов; свойства (имитация поля с помощью функций); управление доступом к полям; метапрограммирование (управление созданием классов, триггеры на создание классов, и др.); полная интроспекция; классовые и статические методы, классовые поля; классы, вложенные в функции и классы), а также функционального программирования (функция является объектом; функции высших порядков; рекурсия; развитая обработка списков; аналог замыканий; частичное применение функции; возможность реализации других средств на самом языке, например, карринг).

## Примеры синтаксиса:

```
def flatten(a):
    """ Flatten list. """
    if a == []:
        return []
    if not isinstance(a, list):
        return [a]
    return reduce(lambda x, y: x + flatten(y), a, [])

def chop(s, size = 1):
    """ Chop list or string. """
    c = s
    r = []
    if size > 0:
        while len(c) > size:
            r.append(c[:size])
            c = c[size:]
        if c != []:
            r.append(c)
    elif size < 0:
        while len(c) > -size:
            r.insert(0, c[size:])
            c = c[:size]
        if c != []:
            r.insert(0, c)
    else:
        raise ValueError("Zero chop size.")
    return r
```

## Книги:

- Плас Дж. В. *Python для сложных задач. Наука о данных и машинное обучение.* // O'Reilly, 2018.
- Horstman C., Necaise R. D. *Python for Everyone.* // Wiley, 2014.
- Necaise R. D. *Data Structures and Algorithms Using Python.* // Wiley, 2011.
- Lenaro G. *Python High Performance Programming.* // Packt Publishing, 2013.
- Lutz M. *Python Pocket Reference. Python in Your Pocket.* // O'Reilly, 2014.
- Mertz D. *Functional Programming in Python.* // O'Reilly, 2015.
- Shaw Z. *Learn Python the Hard Way.* // Addison-Wesley, 2014.

## Сетевые ресурсы:

- *The official home of the Python Programming Language.* <https://www.python.org>
- *Погружение в Python 3.* <https://pep8.ru/doc/dive-into-python-3>
- *Free Interactive Python Tutorial.* <https://www.learnpython.org>

## Консультация по вопросам использования:

[vtasks@jscc.ru](mailto:vtasks@jscc.ru)