

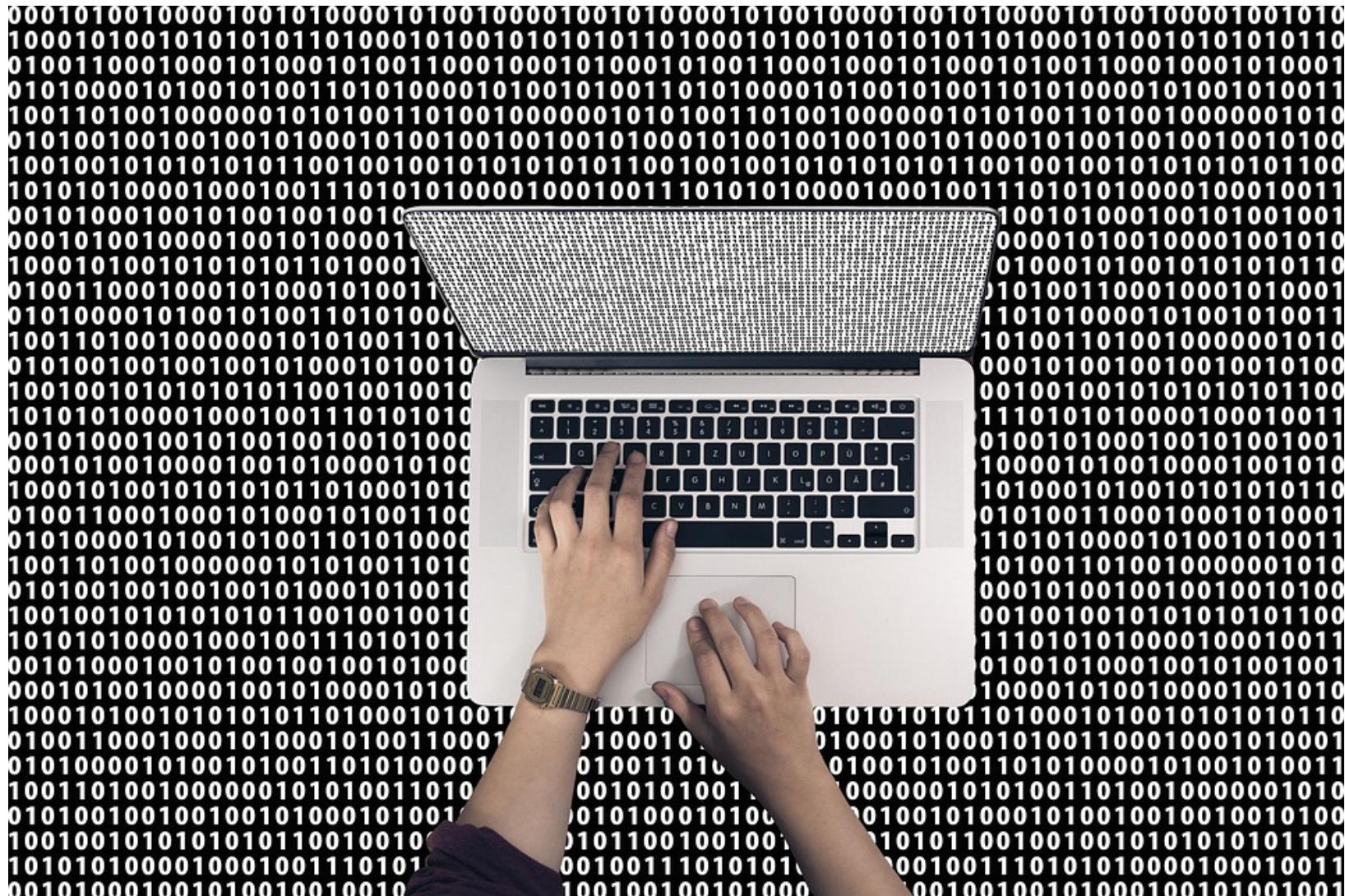
# Principles of Software Programming: Introduction

Svitlana Vakulenko, MSc.

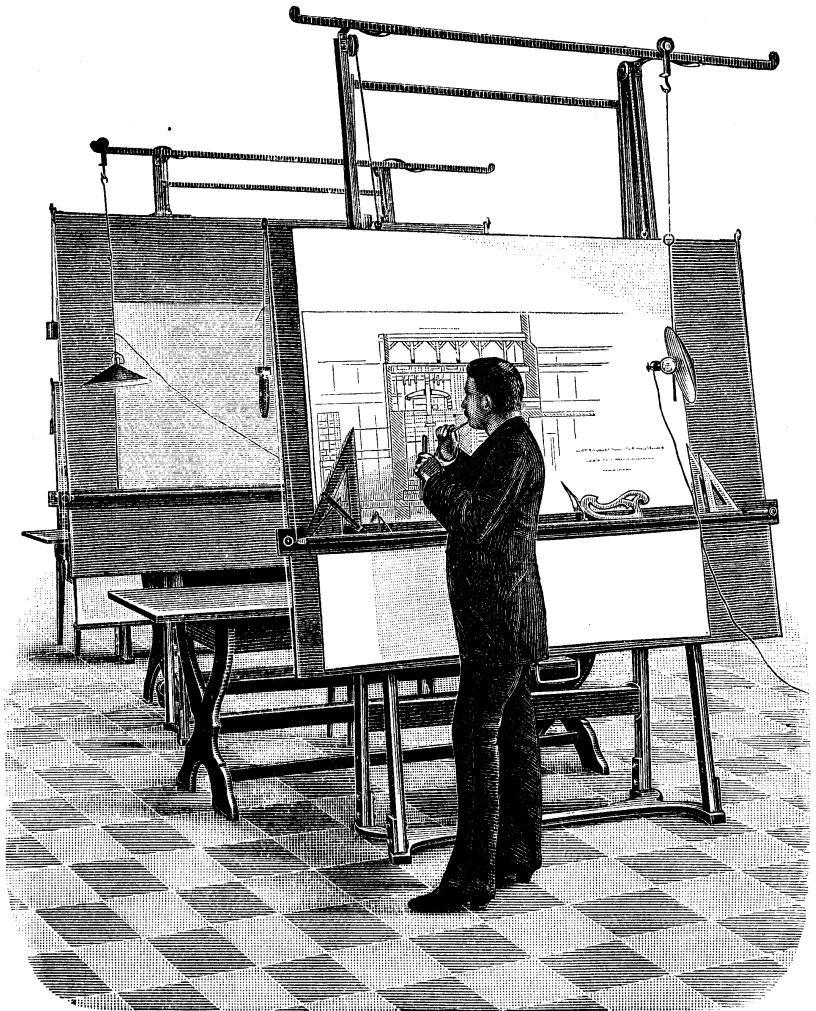
WS 2017

# What is Programming?

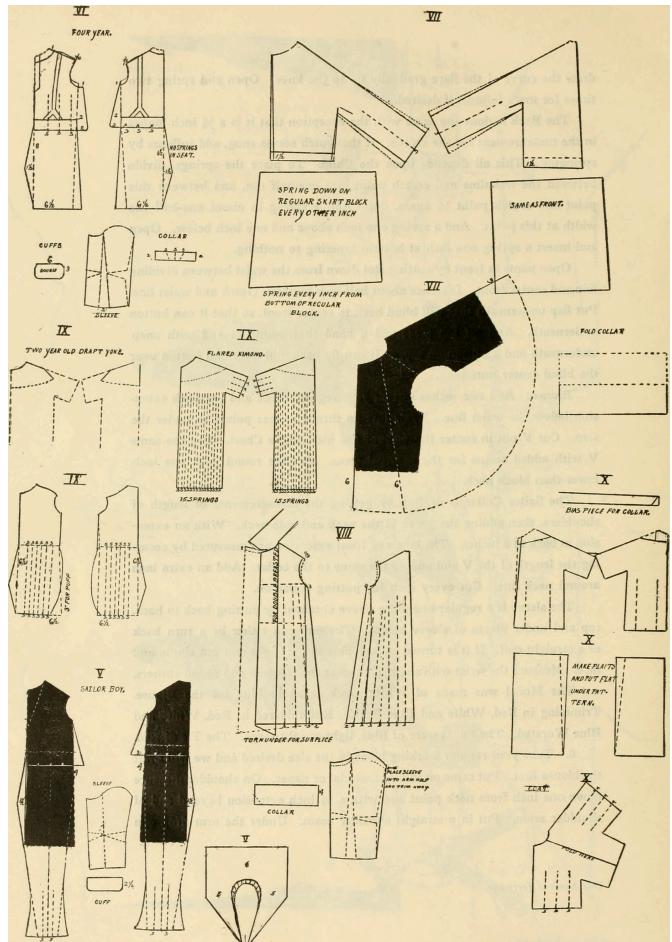
# What is Programming?



# What is Programming?



<https://de.wikipedia.org/wiki/Architekt>



[https://commons.wikimedia.org/wiki/File:Art\\_in\\_dress\\_\(1922\)\\_\(\\_14781442712\).jpg](https://commons.wikimedia.org/wiki/File:Art_in_dress_(1922)_(_14781442712).jpg)

# What is Programming?

Idea: describe a solution to the **problem** in a precise manner, so that computer can follow the **commands**

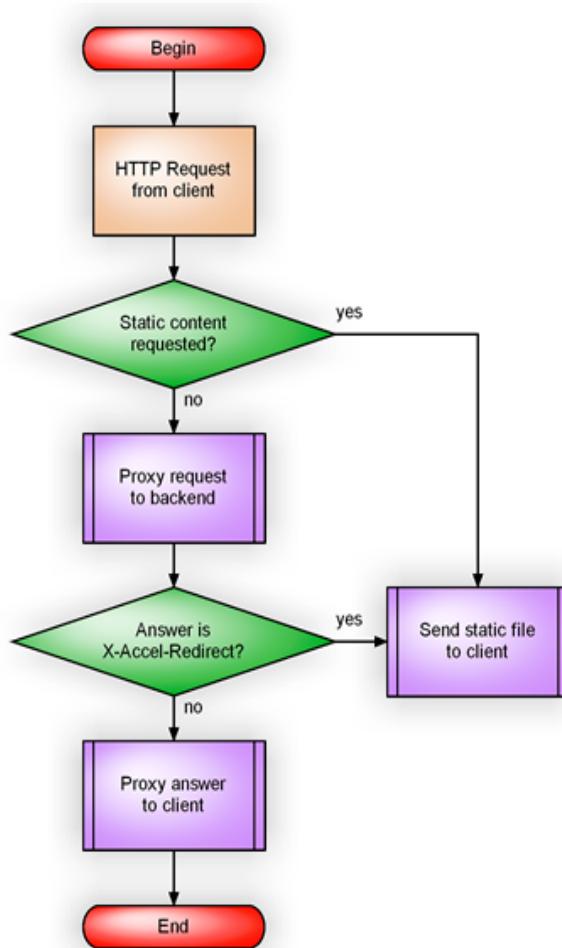
 Problem

 **Creative** solution

Data Inputs + **Program**(Commands) => Results

# Algorithm

## Sequence of commands (computation steps)



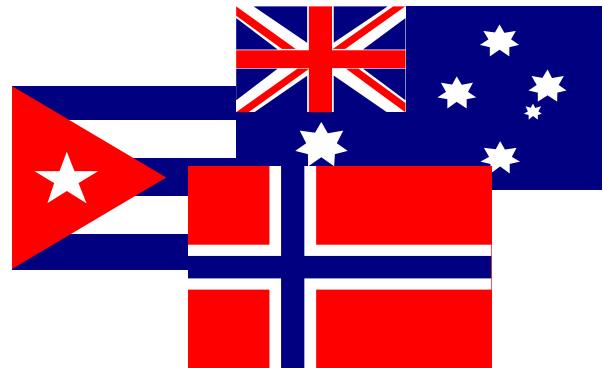
# Examples of Algorithms



- Cooking recipes are algorithms for humans to follow to get a tasty meal
  - „Man nehme 2 dag Butter und erhitze es in einer Pfanne. Inzwischen versprudelt man 3 Eier mit einer Prise Salz. Wenn das Fett heiß ist, gieße man die Eier in die Pfanne. Mit einer Gabel umrühren, bis die Eier stocken, sofort heiß servieren ...“,
- Instructions, Flow chart diagrams
- *Big Bang Theory*: “The Friendship Algorithm”:
  - <http://www.youtube.com/watch?v=k0xgjUhEG3U>

# Languages

- Natural language is complex and ambiguous
- Programming language is simple and straight-forward



# Jupyter Notebook

- Integrated Development Environment (**IDE**)
- **web** application
- **easy-to-use**, interactive
- show **results** in the same document with code
- **presentation**: add description, plots, images & videos

[https://programming.ai.wu.ac.at/1016/notebooks/\\$MATRIKELNUMMER](https://programming.ai.wu.ac.at/1016/notebooks/$MATRIKELNUMMER)

# First program: Hello World!

# First program: Hello World!

```
print("Hello World!")
```

# First program: Hello World!

```
print("Hello World!")
```

```
public class HelloWorld {  
  
    public static void main(String[] args) {  
        // Prints "Hello, World" to the terminal window.  
        System.out.println("Hello, World");  
    }  
  
}
```

<https://excelwithbusiness.com/blog/say-hello-world-in-28-different-programming-languages/>

<https://www.digitalocean.com/community/tutorials/how-to-write-your-first-python-3-program>

<https://introcs.cs.princeton.edu/java/11hello/HelloWorld.java.html>

# Built-in Functions & Parameters

- `abs(x)`
- `pow(x, y)`
- `print(x)`
- `input()`

# Variable assignment

- `abs(x)` ■ `x = -21.9`
- `pow(x, y)` ■ `y = 2`
- `print(x)` ■ `x = "Hello World!"`
- `input()` ■ `s = input()`

# Data Types

- `abs(x)`
  - `x = -21.9` # A floating point
- `pow(x, y)`
  - `y = 2` # An integer
- `print(x)`
  - `x = "Hello World!"` # A string
- `input()`
  - `s = input()` # Return type?

# Type conversion

- explicit:
  - `float(x)`
  - `int(x)`
  - `str(x)`

## Ex.1: Warm-up

- `abs(x)`
- `pow(x, y)`
- `print(x)`
- `input()`

# Type conversion

- explicit:

- float(x)
- int(x)
- str(x)

- implicit:

```
>>> 2.0 + 1
```

```
3.0
```

# Operators

- Arithmetic: +, -, \*, /, %, \*\*, //
- Assignment: =, +=, -=, \*=, /=, %=, \*\*=, //=

## Ex.2: Bart Simpson's Chalkboard



January 14, 1990

[http://simpsons.wikia.com/wiki/Bart\\_the\\_Genius](http://simpsons.wikia.com/wiki/Bart_the_Genius)

# Custom Functions

```
'''A function returning a string and using a local variable'''
def lastFirst(firstName, lastName):
    separator = ', '
    result = lastName + separator + firstName
    return result
print(lastFirst('Benjamin', 'Franklin'))
print(lastFirst('Andrew', 'Harrington'))
```

# Variable scope

- function scope and global scope
- no access to local variables outside the function

```
globalVar = "This is global"  
  
def myFunction():  
    localVar = "This is local"
```

# Pythonic style

- PEP 8 -- Style Guide for Python Code
- Indentation: 4 spaces
- Naming convention: no CamelFont, rather `lower_case_with_underscores`
- Comments:  
`# ??? Why is this here?`  
`"""Create a new user.`  
`Line 2 of comment...`  
`And so on... """`

<https://www.python.org/dev/peps/pep-0008/>

[http://www.voidspace.org.uk/python/articles/python\\_style\\_guide.shtml](http://www.voidspace.org.uk/python/articles/python_style_guide.shtml)

<https://stackoverflow.com/questions/2357230/what-is-the-proper-way-to-comment-functions-in-python>

## Ex.3: Happy Birthday!

Happy Birthday to You

Happy Birthday to You

Happy Birthday Dear **(name)**

Happy Birthday to You.

# Happy Birthday!

```
def happyBirthday(person):  
    print("Happy Birthday to you!")  
    print("Happy Birthday to you!")  
    print("Happy Birthday, dear " + person + ".")  
    print("Happy Birthday to you!")  
happyBirthday('Emily')  
happyBirthday('Andre')
```

# Import

- **Example: math functions**

```
import math  
print (math.sqrt(10))
```

```
from math import pi  
r = 5  
area = pi*r*r
```



[https://commons.wikimedia.org/wiki/File:Artillery-spoked\\_wheel.jpg](https://commons.wikimedia.org/wiki/File:Artillery-spoked_wheel.jpg)

<http://ubuntuforums.org/showthread.php?t=879970>

[https://en.wikibooks.org/wiki/Python\\_Programming/Modules](https://en.wikibooks.org/wiki/Python_Programming/Modules)

# Packages

modules, libraries, extensions

PyPI - Python Package Index is a repository of software

There are currently **120396** packages

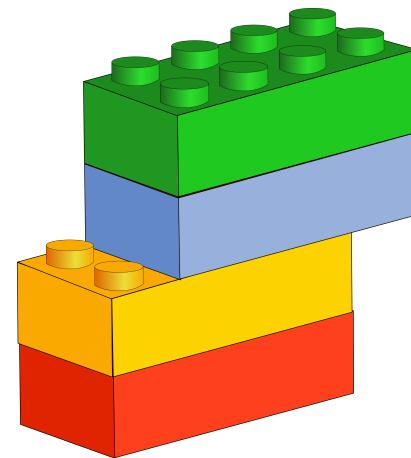
<https://pypi.python.org/pypi>

To use a package from this index: "pip install package"

- **Web data:** requests, scrapy, BeautifulSoup, nltk
- **Data science:** NumPy, SciPy, matplotlib, tensorflow
- **Web development:** django, flask, SQLAlchemy

<https://docs.python.org/3/library/index.html>

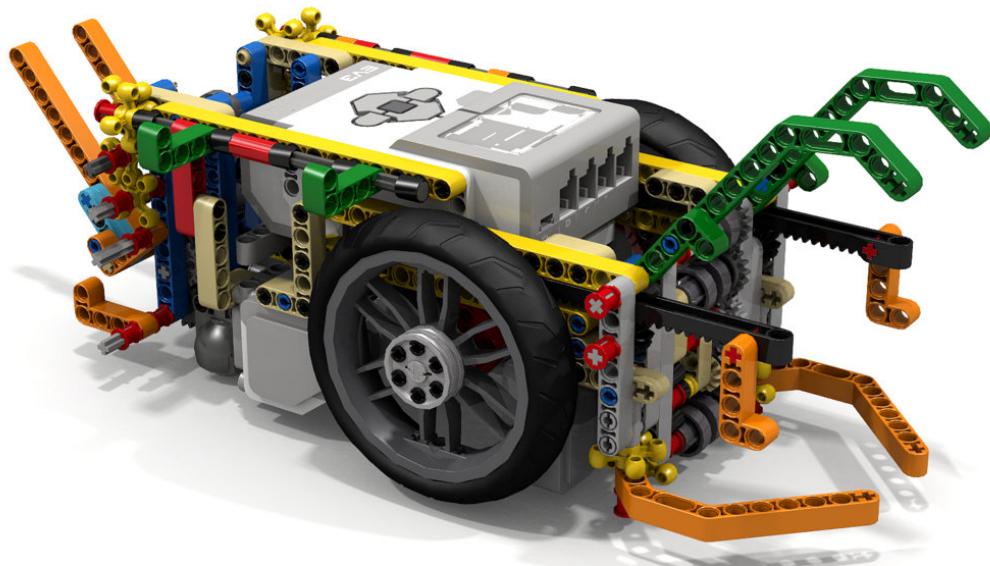
- building blocks
- iterative development
- Reduce, Reuse, Recycle principle



<https://www.goodfreephotos.com/vector-images/lego-blocks-vector-clipart.png.php>

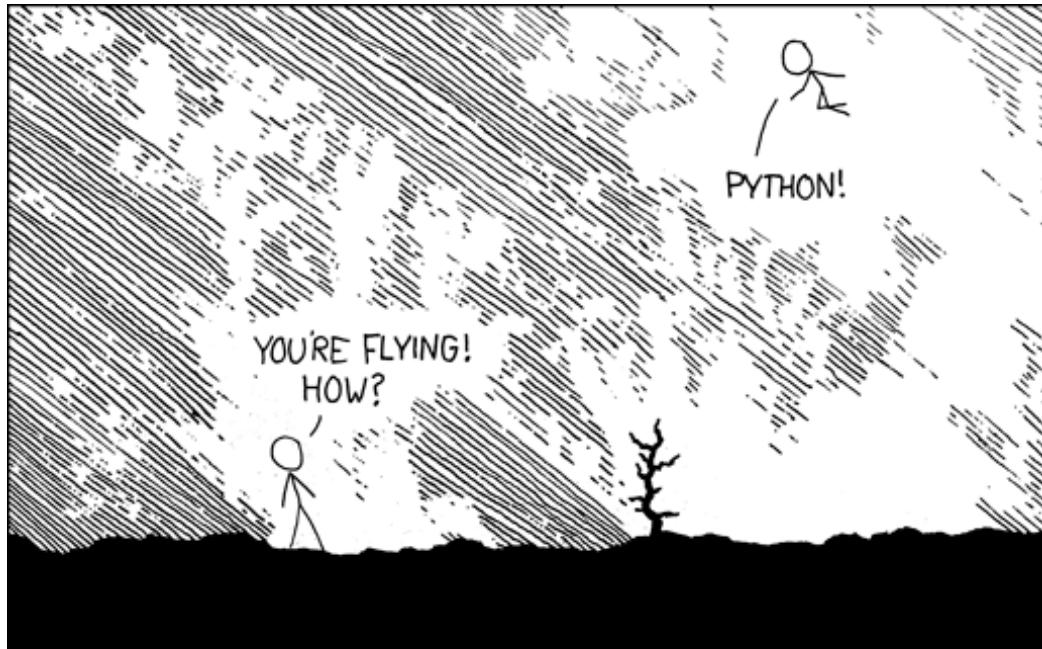


<https://www.flickr.com/photos/keykalou/11931885745>



<https://www.flickr.com/photos/42988571@N08/31195962833>

# import antigravity



## Ex.4: Timer

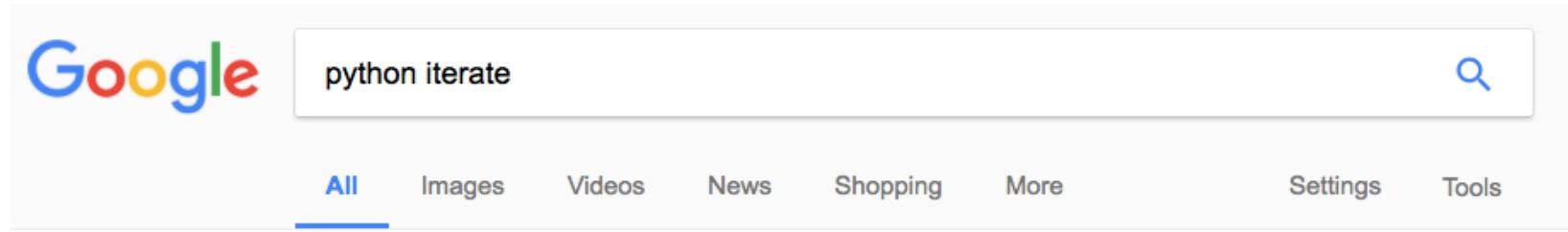


<http://blog.coachseye.com/everything-you-do-makes-a-point/>

# Next Episode

- 3 November 13:00-15:45
- D2.0.031 Workstation-room
- Structured programming in Python
  - Branching (if-else)
  - Iteration (for, while)
- Object Oriented Programming (OOP):
  - Classes
  - Objects
  - Methods
  - Inheritance
  - UML

# Questions?



A screenshot of a Google search results page. The search query "python iterate" is entered in the search bar. The "All" tab is selected, showing approximately 527,000 results. The first result is a link to a Stack Overflow question titled "Iterate a list with indexes in Python - Stack Overflow". Below it is a link to the "ForLoop" page on the Python Wiki. The third result is a link to "Python for Loop Statements" on TutorialsPoint. The fourth result is a link to "6.3. Iterating with for Loops" on Dive Into Python.

python iterate

All Images Videos News Shopping More Settings Tools

About 527.000 results (0,39 seconds)

**Iterate a list with indexes in Python - Stack Overflow**  
<https://stackoverflow.com/questions/126524/iterate-a-list-with-indexes-in-python> ▾  
Sep 24, 2008 - >>> a = [3,4,5,6] >>> for i, val in enumerate(a): ... print i, val ... 0 3 1 4 2 5 3 6 >>> ...

**ForLoop - Python Wiki**  
<https://wiki.python.org/moin/ForLoop> ▾  
Apr 11, 2017 - Contrast the for statement with the "while" loop, used when a condition needs to be checked each iteration, or to repeat a block of code forever.

**Python for Loop Statements - TutorialsPoint**  
[https://www.tutorialspoint.com/python/python\\_for\\_loop.htm](https://www.tutorialspoint.com/python/python_for_loop.htm) ▾  
Python for Loop Statements - Learn Python in simple and easy steps starting from basic to advanced concepts with examples including Python Syntax Object ...

**6.3. Iterating with for Loops - Dive Into Python**  
[www.diveintopython.net › Dive Into Python › Exceptions and File Handling](http://www.diveintopython.net/dive_into_python/exceptions_and_file_handling.html) ▾  
6.3. Iterating with for Loops. Like most other languages, Python has for loops. The only reason you haven't seen them until now is that Python is good at so many ...

# StackOverflow

- specialised Q&A forum

## 6 Answers

active

oldest

**votes**



293



```
>>> a = [3,4,5,6]
>>> for i, val in enumerate(a):
...     print i, val
...
0 3
1 4
2 5
3 6
>>>
```

<https://stackoverflow.com/questions/126524/iterate-a-list-with-indexes-in-python>