



Matthias COLIN

may 2020

matthias.colin@gmail.com

Introduction



- Python project : python.org
- Langage created by Guido van Rossum
 - 1989 : 1st version (0.9)
 - 1994 : version 1.0
 - 2000 : version 2.0
 - **2001** : version 2.1 (Python Software Foundation)
 - **2008** : version 3.0 (non compatible 2.x)
 - 2020 : version 2.7 et 3.8

Assets of the language

- Multi platforms
- Interpreted : `python [-i] monscript.py`
- Simple Syntax
- 3 paradigms of programming
 - Functional `map(sqrt, [1, 4, 9])`
 - Imperative `while delta > epsilon:`
`delta = computeAgain()`
 - Object `valeurs = [3, 5, 7]`
`pos = valeurs.index(7)`
- Rich Integrated Library + External ones (PyPI)
- Big Community

Resources

- Documentation :
 - docs.python.org/2/ ou docs.python.org/3/
 - Tutorial
 - Library Reference
 - Langage Reference
 - Python Module Index
 - Help from interpreter : dir, help, ?
- Python Package Index : PyPI
 - pypi.python.org/pypi
 - 236257 projects
 - outil pip



Environnements Distributions

- Python idle
- IPython : projet SciPy
- Spyder / PyCharm
- Jupyter Notebook
- JupyterLab

IP[y]:



- Anaconda

Language syntax

- Variable
- Base Types
- Block
- Condition
- Loop
- Comprehension
- With

Basic Data Types

bool	True, False	None, 0, 0., [], (), {}, ...
int	3, -3, 0b1001, 0o675, 0x3F	int32/int64 (python 2) infini (python3)
long	9223372036854775808, 4L	la suite des int (python 2)
float	4., 1.5, -7.6E-123 float('nan'), float('inf')	IEEE 754 simple/double
complex	3+4j	
str	'Toto', "Titi"	
datetime.time, datetime.date, datetime.datetime	date(2017,11,20)	
decimal.Decimal	Decimal('1')/Decimal('3')	virgule fixe
fractions.Fraction	Fraction('1/3')	
NoneType	None	

Operators

- Logic : or, and, not
- Comparisons : ==, !=, <, <=, >=, is, is not, in, not in
a is None
3 in [1, 2, 3]
- Numbers : +, -, *, /, //, %, **
- in place version : +=, -=, *=, /=, //=, %=, **=
- Matrix : @
- Bitwise : |, &, ^, ~, <<, >>
- Acces (index, key, slice) : []
s[0], s[-1], s[3:12], s[3:12:2]

Control flow instructions

- if elif else
 - pas de case
- for in
 - « foreach » over all iterable object
 - for i « old school » : range, enumerate
- while
 - no do while
- comprehension : list, dict, generator
- with
 - open/close resource

Functions

- Definition
 - `def f(x):`
 `return x + 1`
 - `lambda x: x+1`
 - 2^e order : `map`, `iter`, `all`, `any`, `filter`
- Argument
 - position or keyword
 - variable args / tuple / dict
- Return value / None
- Scope of variables
- Built-in functions

Functions (II)

- Mathematical :

functions Built-In : float, int, long, abs, cmp, min, max, sum

module math (floor, sqrt, cos, pi, e, ...)

module statistics (mean, median, ...)

modules random, cmath, decimal, fractions

scientific python : numpy

Strings :

functions Built-In : len, str, repr, cmp

methods : join, upper, lower, index, format, ...

Objets Standards

- Strings : str
- Lists : list
- Tuples : tuple
- Dictionnaires : dict
- Generators

Sequences and Dictionnaires

- Listes : list
[1,2,3], [3],[], [[1,2,3], [4,5,6]]
- Tuples : tuple
1,2,3, (1,2,3), (1,), ()
- Sets : set
{1,2}
- Dictionnaires : dict
{'Pau':64, 'Toulouse':31}
- Operators : + et [] (accès or slice)

Iterable/Iterator/Generator

- Un itérateur permet de parcourir une donnée complexe
 - Built-In fonction `next()`
- Un objet itérable renvoie un itérateur sur lui-même
 - Built-In fonction `iter()`
- Permet un parcours avec une boucle, une comprehension list

```
spam = ['eggs1', 'eggs2', 'eggs3']
for item in spam:
    print item
```
- Un générateur fournit des valeurs à la demande
 - Faible coût mémoire
 - Un générateur est itérable
 - Implémentation avec `yield` et `yield from (*)`
 - Exemple : `range(10)`

(*) python 3 uniquement

Package/Module

- Déclaration et structure
- Convention de nommage
- Opérations sur les modules

Programmation Orientée Objets

- Concepts de la POO
- Membres d'instances et de classes
- Méthodes spéciales
- Encapsulation

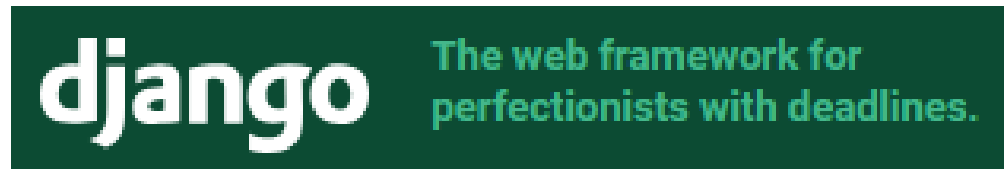
Librairies Communes

- Système / processus : sys
- Système de fichiers : os.path, pathlib, glob
- Expressions régulières : re
- Base de données : PEP249

Gestion des Fichiers

- Ouverture/fermeture de fichiers
- Lecture/Ecriture
- Informations sur les fichiers
- Gestion des répertoires

Programmation Web



IA



TensorFlow



scikit
learn

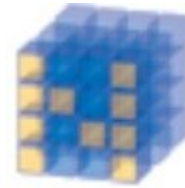


Keras

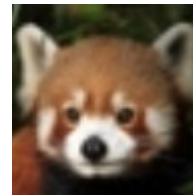
- <https://www.tensorflow.org/>
- <https://scikit-learn.org>
- <https://keras.io/>
- <https://github.com/Microsoft/cntk>
- <https://github.com/Theano/Theano>

Calcul Scientifique en Python

- SciPy.org
 - NumPy : numpy.org
 - SciPy
 - Matplotlib : matplotlib.org
 - IPython : ipython.org
 - SymPy : sympy.org
 - Pandas : pandas.pydata.org



IP[y]:

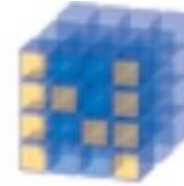


Environnement IPython

IP[y]:

- <http://ipython.org>
- Shell python
- Interactivité ++
- Aide
- Complétion automatique

NumPy



- <http://www.numpy.org/>
- Types NumPy :
- N-dimensionnal array + matrix
 - Broadcasting
- Algèbre Linéaire
- FFT
- Finances
- Input/Output
- Polynomes
- Tris
- Statistiques

Types Numpy

- <https://docs.scipy.org/doc/numpy-1.13.0/user/basics.types.html>
- Taille + Signe
- Exemple : int8, uint32
- Entiers
- Flottants
- Complexes

Matplotlib & Basemap

- `matplotlib.pyplot`
- `mpl_toolkits.basemap`

