

Séminaire Python Débutant

Yoann Beaugnon

27 octobre 2021

Introduction

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

**Python c'est
quoi ?**

Objectif de ce
Séminaire
Philosophie

Bases

Nombres
Fonction
String
Control Flow
Listes
Dictionnaires
raise

Outils

pip/venv
pytest
tox



Objectifs

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

**Objectif de ce
Séminaire**

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

Objectif

- (Re)introduire les bases de python.
- Montrer et encourager l'utilisation de types et méthodes fournis par python.
- Introduire des outils pour plus facilement gérer ses projets et la clareté de ceux ci

1 Introduction

- Python c'est quoi ?
- Objectif de ce Séminaire
- Philosophie

2 Bases

- Nombres
- Fonction
- String
- Control Flow
- Listes
- Dictionnaires
- raise

3 Outils

- pip/venv
- pytest
- tox

Philosophie

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

```
import this
```

Nombres

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire
Philosophie

Bases

Nombres

Fonction
String
Control Flow
Listes
Dictionnaires
raise

Outils

pip/venv
pytest
tox

Nombres

- `int : i = 42; c = 1 + 2j`
- `float : f = 6.62e-34`
- `supporte les opérations élémentaires ; + - / * **`

Fonction

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Lists

Dictionnaires

raise

Outils

pip/venv

pytest

tox

Fonction

```
def addition(a,b):  
    '''Ajoute deux nombres'''  
    c = a + b  
    return c # Renvoie c.  
addition(1+1)
```

Fonction lambda

```
f = lambda x : 2*x + 1  
f(2)
```

Fonction

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire
Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

arguments optionels

```
def addition(a,b,c=0):  
    return a + b + c  
addition(1,1)  
>>> 2  
addition(1,1,1)  
>>> 3  
addition(1,1,c=1)  
>>> 3
```

Fonction utilisant self

Certaines fonctions ont un argument self, elles s'appliquent sur une variable ex : `(2 + j).conjugate()`

String

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

str

- déclaration : `s = "Hello, World!"`
- contatenation : `s = 'Hello' + ', ' + 'World!'`
- multiplication : `s = "Hello " * 5`
- affichage : `print("Hello World!")`

Format

Outils intégrés à python (format)

- f string : `a = "toto"; f"Mon nom est : {a}"`
- .format : `a = "toto"; "Mon nom est : {}".format(a)`

String

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

Méthodes supplémentaires

- `s = "Hellu Wurld".replace("u", "o")`
- `s = "0123456".find("34")`
- Plein d'autres méthodes, regardez dans la documentation avant de réimplémenter.

if/else

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire
Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

if/elif/else

```
if a < 0 :  
    print("a est positif")  
elif a == 0:  
    print("a = 0")  
else :  
    print("a est positif")
```

While

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

While

```
i = 0
while i < 10:
    if i == 6:
        break
    i = i + 1
else:
    print("6 n'est pas entre 0 et 9")
```

For

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire
Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

For

```
for i in range(0, 10):  
    if i == 6:  
        break  
else:  
    print("6 n'est pas entre 0 et 9")
```

Liste

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

list

```
l = ['zero', 1, 'deux', ['cookie', 'gateau']]
```

slicing

```
l[2]
>>> 'deux'
l[-1]
>>> ['cookie', 'gateau']
l[1:]
>>> [1, 'deux', ['cookie', 'gateau']]
l[-1][1]
>>> 'gateau'
```

Méthodes de bases

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

méthodes

- `.append(a)`
- `.remove(a)`
- `.insert(i, a)`
- `.index(a)`

tester si un élément est dans la liste

```
if 1 in [1,2,3,4] :
```

Liste

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire
Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

contatenação

- `a = [0, 1]; b = [2, 3]; c = a + b`
- `d = a * 2`

copie partielle

```
a = [0, 1]
b = a
a[0] = 1
b
>>> [1, 1]
```

Il existe une méthode permettant de créer une liste ayant les mêmes éléments : `.copy()`

Liste

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

copy

```
a = [0, [0, 1]]
b = a.copy()
a[0] = 1
b
>>> [0, [0, 1]]
a[1][0] = 1
b
>>> [0, [1, 1]]
```

Listes par compréhensions

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

Listes par compréhensions

```
carres = [i**2 for i in range(4)]
carres
>>> [0, 1, 4, 9]
```

Possibilité d'imbruquer des listes par compréhensions

```
l = [[0, 1], [2, 3]]
flat = [cell for line in l for cell in line]
flat
>>> [0, 1, 2, 3]
```

Unpacking

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire
Philosophie

Bases

Nombres

Fonction

String

Control Flow

Lists

Dictionnaires

raise

Outils

pip/venv

pytest

tox

Tuple

```
def inverser(a, b):  
    return (b, a)
```

(Un)packing

```
a, b = (a, b)  
c = (a, b)  
print(a); print(c[0])  
print(b); print(c[1])
```

```
d = [*c]  
inverser(c[0], c[1])  
inverser(*c)
```

*args

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire
Philosophie

Bases

Nombres

Fonction

String

Control Flow

Lists

Dictionnaires

raise

Outils

pip/venv

pytest

tox

Faire passer un nombre indéfini d'argument à une fonction

```
def maximum(*args):  
    m = args[0]  
    for r in args[1:]:  
        if r > m:  
            m = r  
    return m  
maximum(1,3,8,5)  
>>> 8
```

Dictionnaires

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire
Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

bases

```
dictionnaire = {"un" : 1, 2 : "deux"}  
d["un"]  
>>> 1  
d["trois"] = 3
```

méthodes

- `.values()`
- `.keys()`

(Un)packing

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

*

```
d = {'un' : 1, 'deux' : 2}
print(d)
>>> {'un': 1, 'deux': 2}
print(*d) :
>>> un deux
```

**

```
def aire(a=0, b=0):
    return a*b
d = {'a':2, 'b':3}
aire(**d)
>>> 6
```

**kwargs

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire
Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

passer un nombre d'argument optionels indéfini à une fonction

```
def afficher_dict(**kwargs):  
    for key in kwargs:  
        print(key, kwargs[key])
```

raise

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

Lever une erreur

```
a = 2 if type(a) is not str : raise ValueError("Erreur, n'est pas  
une str".format(a))
```

Un gestionnaire de packet pour python

Inclut par défaut depuis python 3.4

- `python -m pip install numpy >= 1.21`
- `python -m pip list`
- `python -m pip uninstall numpy`

virtual environnement

- `python -m pip install virtualenv`
- `python -m venv env`
- `source env/bin/activate`
- `deactivate`

faciliter le partage du code

```
pip install -r requirements.txt
```

requirements.txt

```
numpy >= 1.21  
matplotlib >= 3.4.3
```

test

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

pytest

```
python -m pip install pytest  
pytest
```

test_monprogramme.py

```
def test_addition():  
    assert 1+1 == 2
```

tox

Séminaire
Python
Débutant

Yoann
Beaugnon

Introduction

Python c'est
quoi ?

Objectif de ce
Séminaire

Philosophie

Bases

Nombres

Fonction

String

Control Flow

Listes

Dictionnaires

raise

Outils

pip/venv

pytest

tox

tox

```
python -m pip install tox
```

tox.ini

```
[tox]
envlist = py38, py39, linters
skipsdist = true

[testenv]
deps = pytest
      -r{toxidir}/requirements.txt
commands = pytest

[testenv:linters]
deps = flake8
commands = flake8

[flake8]
exclude = .git .pytest_cache __pycache__
```