

# Cours de Programmation en Python

Mendy Fatnassi

10 décembre 2020

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Présentation . . . . .	3
1.2	Installation bibliotheque python . . . . .	3
1.3	Python Generalité . . . . .	3
<b>2</b>	<b>variable</b>	<b>4</b>
2.1	Entree/Sortie . . . . .	4
<b>3</b>	<b>Dictionnaire</b>	<b>5</b>
<b>4</b>	<b>Structure de controle</b>	<b>6</b>
4.1	Operateur de controle . . . . .	6
4.2	Operateur de boucle . . . . .	6
<b>5</b>	<b>Les fonctions</b>	<b>7</b>
5.1	Definir une fonction . . . . .	7
<b>6</b>	<b>Biblotheque Numpy/Scipy/Matplotlib</b>	<b>8</b>

# Chapitre 1 : Introduction

## 1.1 Présentation

Python est un langage de programmation interprété, c'est-à-dire que les instructions que vous lui envoyez sont « **transcrites** » en langage machine au fur et à mesure de leur lecture.

D'autres langages (comme le C / C++) sont appelés « langages compilés » car, avant de pouvoir les exécuter, un logiciel spécialisé se charge de transformer le code du programme en langage machine.

On appelle cette étape la « compilation ».

## 1.2 Installation bibliothèque python

Depuis python 3.4 l'utilitaire pip est installé avec la package d'installation, sinon télécharger le : `$sudo aptget install pip`

Une fois cela fait utilisé pip pour télécharger les bibliothèque python : `$sudo pip install nom_biblio`

## 1.3 Python Generalité

Comme pour un script sh, python a un shebang `#!/usr/bin/python`

.

Pour lancer un l'interpreteur python depuis le terminal tapez : `pythonX.X` (python3.5).

Pour lancer directement un script depuis le terminal : `+pythonnom_script.pyOU./nom_script.py+`

# Chapitre 2 : variable

Python comme dit précédemment n'est pas un langage typé c-a-d qu'il suffit de déclarer une variable sans son type et de l'affecter directement à la valeur souhaitée.

exemple :

```
varInt = 12
```

```
varFloat = 12.5 varString = "StringQuelconque"
```

une chaîne peut être entourée soit de guillemets ou d'apostrophe mais il faudra échapper les caractères guillemet/apostrophe si celui-ci figure dans la chaîne.

Si on ne veut pas échapper les caractères on peut mettre la chaîne en triple guillemet `"""chaîne d'un caractère"""`.

## 2.1 Entree/Sortie

`type(nom_var)` => renvoie le type de la variable.

`print("a = ", a, " et b = ", b)` => affiche la variable ou le texte sur la sortie standard.

`input("message : ")` => saisie utilisateur.

Rq : si on veut stocker le résultat, `varRes = input("entrez resultat : ")`.

# Chapitre 3 : Dictionnaire

En python il existe des tableau associatif appelé dictionnaire et se declare de deux facons :

soit avec la class *dict()* : *mon\_dic = dict()*

ou alors avec des parenthese : *dictionnaire = Cles : valeur*

Les parenthèses () délimitent les tuples, les crochets [] délimitent les listes et les {} délimitent les dictionnaires.

Exemple :

```
#Declaration du dictionnaire
```

```
d={"salut" : "hi" , "Montagne": "Mountain" , 2: "Wave" , "Manger": "eat", "Boire": "Drink"}
```

```
#Afficher tout le dictionnaire
```

```
print("\n Valeur du dictionnaire =", d, "\n")
```

```
#Affiche
```

```
print("Salut --->", d["salut"], "\n")
```

## Ajout

Pour ajouter une valeur dans un dictionnaire on place directement la cles et

ca valeur dans le dictionnaire : *d["ajouter"] = "add"*

```
d = "salut" : "hi", "Montagne" : "Mountain", 2 : "Wave", "Manger" : "eat", "Boire" : "Drink", "ajouter" : "add"
```

## Supression

On peut utiliser la methode *del()* : *delnomDict[cles] -- > deld["salut"]*

```
d = "Montagne" : "Mountain", 2 : "Wave", "Manger" : "eat", "Boire" : "Drink", "ajouter" : "add"
```

# Chapitre 4 : Structure de controle

## 4.1 Operateur de controle

Syntaxe :

```
1 if condition :
2     print("Text")
3 elif condition :
4     print("Text")
5 else :
6     printf("Text")
```

On peut utiliser les mot clé **OR,AND,NOT** dans l'expression de la condition. On peut aussi utiliser **in** pour savoir si la variable appartient a cette sequence ou non.

Exemple : if varString in "abcdef" => (*sivarstring* = a|b|c|d|e|f)

## 4.2 Operateur de boucle

```
1 while condition :
2
1 for element in sequence :
2
```

La variable "element" est créée par le for; ce n'est pas à nous de l'instancier. Elle prend successivement chacune des valeurs figurant dans la séquence parcourue sequence:

# Chapitre 5 : Les fonctions

## 5.1 Définir une fonction

Création d'une fonction :

```
1 def nom_de_la_fonction(parametre1, parametre2, parametre3,  
2     parametreN):
```

Si un parametre n'est pas saisie a l'appel d'une fonction , on peut donner une valeur par défaut a ce parametre par exemple :

```
1 def table(nb, max=10):  
2  
3     i = 0  
4     while i < max:  
5         print(i + 1, "*", nb, "=", (i + 1) * nb)  
6         i += 1
```

On peut appeler la fonction de deux façons : soit en précisant le numéro de la table et le nombre maximum d'affichages, soit en ne précisant que le numéro de la table (table(7)).

Dans ce dernier cas,max vaudra 10 par défaut.

# Chapitre 6 : Biblotheque Numpy / Scipy / Matplotlib

Ces 3 bibliotheque permette d'utiliser des fonction mathematique , traces des graphes , faire des probabilite etc ...