



# Initiation au langage ‘Python’

---

JEAN DIDIER KOUAKOU

[jeandidikouakou@gmail.com](mailto:jeandidikouakou@gmail.com)

# Initiation au langage python

## C'est quoi python ?

---

Python est un langage de programmation (au même titre que le C, C++, fortran, java . . .), développé en 1989. Ses principales caractéristiques sont les suivantes :

- ❖ «open-source» : son utilisation est gratuite et les fichiers sources sont disponibles et modifiables ;
- ❖ Simple et très lisible ; doté d'une bibliothèque de base très fournie ;
- ❖ Importante quantité de bibliothèques disponibles : pour les calculs scientifiques, les statistiques, les bases de données, la visualisation, le traitement de données, l'intelligence artificielle . . . ;
- ❖ Grande portabilité : indépendant vis à vis du système d'exploitation (linux, windows, MacOS) ;
- ❖ Etc.

# Initiation au langage python

## Utilisations de Python et librairies

---

- ❖ Programmation web : Django, Zope, Plone, . . .
- ❖ Bases de données : MySQL, Oracle, . . .
- ❖ Réseaux : TwistedMatrix, PyRO, VTK, . . .
- ❖ Représentation graphique : matplotlib, VTK, . . .
- ❖ Calcul scientifique : numpy, scipy, . . .

# Initiation au langage python

## Table de matière

---

Chap. 1 : Installation de Pycharm & Python;

Chap. 2 : Premier programme en langage python

Chap. 3 : Variables et Opérateurs

Chap. 4 : Les entrées au clavier

Chap. 5 : Instructions Conditionnelles

Chap. 6 : Instructions Itératives

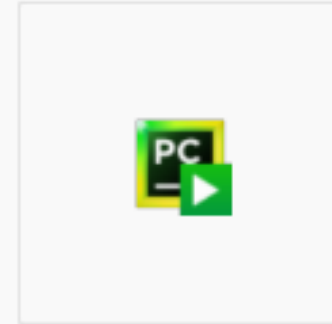
Chap. 7 : Mon premier jeu avec le langage Python

# Initiation au langage python

## Chap. 1 : Installation de Pycharm & Python;



Python



Pycharm

# Initiation au langage python

## Chap. 2 : Premier programme en langage python ;

---

Pour commencer à écrire un programme, il faudra avoir quelques instructions de départ.

### 1. Insertion d'un commentaire

C'est une partie du code qui n'est pas interpréter par la machine. En python pour insérer un commentaire, on utilise le symbole `"# "` suivi du commentaire

Exemple : `# Ceci est un commentaire sur une ligne en python`

### 2. Affichage dans la console

Pour afficher les informations liées au programme, on utilise l'instruction ***print***

**Syntaxe :** `print(" l'information à afficher ")`

Exemple : `print (" J'aime coder en python ")`

### 3. Cas pratique

# Initiation au langage python

## Chap. 3 : Variables et Opérateurs ;

---

### 1. Utilisation d'une variable

- ❖ Une variable est un espace mémoire réservé à une inconnue. Elle est utilisée pour stocker une valeur qui peut être de différentes type ; soit un nombre, une chaîne de caractère, un booléen, ...
- ❖ Une variable est identifiée par un nom. Ce nom doit commencer par une lettre et ne peut contenir ni espace, ni caractères spéciaux
- ❖ Pour affecter une valeur à une variable (c'est-à-dire stocker une valeur dans une variable), on utilise le symbole = suivi de la valeur.

### 2. Types d'une variable

- ❖ entiers (32 bits), Type : int
- ❖ réels (64 bits), type : float. Exemples de valeurs : 4. ; 5.1 ; 1.23e-6
- ❖ complexes type : complex. Exemples de valeurs : 3+4j ; 3+4J
- ❖ booléens type : bool. Exemples de valeurs : True ; False.

# Initiation au langage python

## Chap. 3 : Variables et Opérateurs ;

---

### 3. Opérateurs de base

— affectation

```
>>> i = 3 # i vaut 3
```

```
>>> k=r=2.15
```

```
>>> x=complex(3,4)
```

### 4. Opérations de base

— Opérateurs addition, soustraction, multiplication, division +, -, \*, \.

— Opérateurs puissance, valeur absolue \*\*, pow, abs, \dots

— Opérateurs de comparaison ==, is, !=, is not, >, >=, <= Opérateurs logiques or, and, not

— Conversion : >>> int(3.1415) correspond á 3.



# Initiation au langage python

## Chap. 3 : Variables et Opérateurs ;

---

### 5. Chaines de caracteres

(string en anglais) est un objet de type, str.

```
>>> "je suis une chaine" ou, >>> 'je suis une chaine'  
>>> 'pour prendre l\'apostrophe' ou >>> "pour prendre l\'apostrophe"  
>>> " " "ecrire sur plusieurs lignes" " " ou 'ecrire\nsur\nplusieurs\nlignes'
```

### 6. Concaténation

On peut mettre plusieurs chaînes de caractères bout à bout avec l'opérateur binaire de concaténation, noté +.

```
>>> s = 'i vaut'  
>>> i = 1  
>>> print( s+i ) >>> print( s + ' ' + str(i))
```

# Initiation au langage python

## Chap. 3 : Variables et Opérateurs ;

---

### 7. accès aux caractères

Les caractères qui composent une chaîne sont numérotés à partir de zéro. On peut y accéder individuellement en faisant suivre le nom de la chaîne d'un entier encadré par une paire de crochets :

```
>>> "bonjour"[3]. Rep ; 'j'. Aussi, "bonjour"[2:]; "bonjour"[:3]; "bonjour"[3:5]. Rep ; 'njour' 'bon' 'jo'
```

NB : En python ;

\ désigne une division décimale

: la division normale qu'on connaît.

\\ désigne une division entière c'est-à-dire retourne que la partie entière de la division.

% retourne le reste de la division

\*\* est utilisé pour faire les calculs de puissance ;

# Initiation au langage python

## Chap. 4 : Les entrées au clavier;

---

Dans ce chapitre, nous allons voir les instructions à utiliser pour récupérer et stocker une donnée dans une variables.

### 1. Instruction à utiliser

❖ On utilise l'instruction *input*.

**Exemple :** `name = input("Entrez votre nom : ")`

NB : cette instruction ne permet que de stocker une chaine de caractère. La variable phrase va contenir une chaine de caractère grâce au symbole `=` qui désigne une affectation.

❖ Pour stocker un autre type de variable, on utilise l'instruction *input* puis on fait une conversion dans le type de donnée adapté.

**Exemple :** `age = int(input("Entrez votre age : "))`    *#En considérant que l'age á saisir est un entier*

`moyenne = float(input("Entrez votre moyenne : "))`    *#En considérant que la moyenne á saisir est un réel*

❖ L'instruction, `moyenne = eval(input("Entrez votre moyenne : "))` permet d'évaluer l'opération que l'utilisateur saisira.

# Initiation au langage python

## Chap. 5 : Les instructions conditionnelles

---

### 1. L'indentation

Les fonctions Python n'ont pas de begin ou end explicites, ni d'accolades qui pourraient marquer là où commence et où se termine le code de la fonction. Le seul délimiteur est les deux points (« : ») et l'indentation du code lui-même. Les blocs de code (fonctions, instructions if, boucles for ou while etc) sont définis par leur indentation. L'indentation démarre le bloc et la désindentation le termine. Il n'y a pas d'accolades, de crochets ou de mots clés spécifiques. Cela signifie que les espaces blancs sont significatifs et qu'ils doivent être cohérents.

Voici un exemple :

ligne d'en-tête

première instruction du bloc

...

dernière instruction du bloc

# Initiation au langage python

## Chap. 5 : Les instructions conditionnelles

---

### 2. Contexte

Dans le déroulement d'un algorithme, certaines instructions sont exécutées que si une condition a été satisfaite auparavant. La condition est une comparaison entre deux valeurs du même type.

- ❖ En langage mathématique, les opérateurs de comparaison sont :  $=$  ,  $\neq$  ,  $<$  ,  $>$  ,  $\leq$  et  $\geq$
- ❖ En langage Python, ces opérateurs de comparaison s'écrivent respectivement : `==` , `!=` , `<` , `>` , `<=` et `>=`

### 3. Instruction conditionnelle du type : Si... alors...

En langage Python, l'instruction s'écrit : ***if condition*** :

Le bloc d'instruction qui suit le if sera placé en dessous et décalé d'une tabulation (en général 3 ou 4 caractères). On dit que les instructions sont indentées.

**Exemple :**

```
a = -150
if a < 0:
    print('a est negatif')
```

# Initiation au langage python

## Chap. 5 : Les instructions conditionnelles

---

### 3. Instruction conditionnelle du type : Si... alors...

En langage Python, l'instruction s'écrit : **if condition** : Le bloc d'instruction qui suit le if sera placé en dessous et décalé d'une tabulation (en général 3 ou 4 caractères). On dit que les instructions sont indentées.

Exemple :

```
a = -150
if a < 0:
    print('a est negatif')
```

La condition peut comporter plusieurs tests.

- ❖ **if (condition1) and (condition2)** vraie, signifie que la condition1 est vraie et que la condition2 est vrai. Les deux conditions doivent obligatoirement être vraies pour que le test soit vrai.
- ❖ **if (condition1) or (condition2)** vraie, signifie que la condition1 est vraie ou que la condition2 est vrai ou que les deux conditions sont vraies.

Exemple ;

```
A = float(input("Veuillez entrer la valeur de A : "))
if A<0 and A!=0 :
    print(" L'ensemble de definition est ]-infini; 0[")
```

# Initiation au langage python

## Chap. 5 : Les instructions conditionnelles

---

### 4. Instruction conditionnelle du type : Si... alors...sinon

En langage python, l'instruction s'écrit : En reprenant l'exemple précédent, on peut ajouter ;  
if condition:

    instructions

else :

    instructions

### 5. Instruction conditionnelle du type : Si... alors...sinon si... alors... sinon

if condition :

    instructions

elif condition :

    instructions ...

else :

    instructions

**Exercice :** Ecris un programme python qui demande d'entrer une note supérieure à 10 et donne par la suite la mention correspondante à cette note.

# Initiation au langage python

## Chap. 6 : Les instructions itératives ;

---

Les instructions itératives sont aussi appelées les boucles. Pour cette initiation, nous allons étudier *la boucle for* et *la boucle while*.

### 1. La boucle for

Pour ... allant de ... à... faire ; ainsi peut être décrite la boucle for.

- ❖ Ce type de boucle est appelé boucle bornée. En effet, le nombre d'itération (répétitions) est connu à l'avance.
- ❖ On utilise donc un compteur qui, une fois atteint une certaine valeur, arrête l'exécution de la boucle.
- ❖ A chaque passage dans la boucle, le compteur sera incrémenté (avancé de +1) ou décrémenté (soustraire de -1) selon le cas. En langage python, l'instruction (la syntaxe) s'écrit :

```
for ... in range (debut, fin, pas) : #Le debut, la fin et le pas sont des nombres relatifs
    instructions
```

ou bien

```
for ... in range (debut, fin) :      #Le pas est égale á 1.
    instructions
```

ou bien

```
for ... in range (fin) :            #Le pas est égale á 1 et le debut est 0.
    instructions
```



# Initiation au langage python

## Chap. 6 : Les instructions itératives ;

---

### 2. La boucle while

**Tant que... faire** ; ainsi peut être décrite la boucle while.

Avec ce type de boucle, on répète l'exécution tant qu'une condition préalable définie est satisfaite. Le test s'effectue en début de boucle, à chaque passage. Ce type de boucle est appelée boucle non bornée ou **boucle conditionnelle**. En effet, le nombre d'itération (répétitions) n'est pas connu à l'avance.

while ... condition :

    instructions

**Exercice 1** : Un programme qui liste les nombres de 0 à 100 par pas de 10.

**Exercice 2** : Déclarez une variable qui stock un mot de passe de votre choix

1. Demander à un utilisateur de s'identifier à travers le mot de passe
2. Affiche un msg d'encouragement pour le mdp trouvé
3. Après 03 tentatives affiche que le code est erroné

*Keep learning, look for skills not  
for marks !*

JEAN DIDIER KOUAKOU

jeandidikouakou@gmail.com