

1 Python ?

Python est un langage de programmation puissant, riche de multiples possibilités, et portable ; c'est-à-dire que l'on peut l'utiliser sur de multiples plateformes (Windows, linux, ...) pour y réaliser des applications très variées allant de simples programmes jusqu'à des projets très complexes incluant les graphismes, le multimédia, les communications, le web, l'univers des jeux, ... Malgré tout, Python reste un langage accessible doté d'une syntaxe relativement intuitive qui facilite son apprentissage.

A la différence de langages tel que le « C », Python est un langage interprété : les instructions sont transcrites en langage machine au fur et à mesure de leur lecture par « l'interpréteur python ».

Python est libre de droit et bénéficie du travail d'une vaste communauté internationale : de nombreux exemples sont disponibles ainsi que des bibliothèques qui simplifient le travail du développeur.



Tout le long de ce cours, vous pourrez vous appuyer sur le site « **Introduction au langage de programmation Python3** » qui complète et approfondit les notions abordées ici !

2 Prise en main

Nous utiliserons Python avec l'environnement de développement **IDLE** qui est intégré à Python et propose un certain nombre d'outils :

- Un éditeur de textes pour écrire le programme ou script.
- Un interpréteur pour exécuter le programme.
- Un débogueur pour tester le programme.

2.1 La gestion des espaces

En Python, on peut sauter des lignes pour plus de clarté, ce n'est pas pris en compte lors de l'exécution du programme. Par contre, on ne peut pas ajouter d'espace en début de ligne : on appelle ce décalage « **indentation** » ; en Python, il permet de délimiter un bloc d'instructions.

Exemple :

<p>Bloc « SI »</p>	{	<pre>>>> a=5 >>> if a>0: print("a est positif") a est positif ...</pre>	# teste si un nombre est positif	<p>Les commentaires sont précédés par #</p>
------------------------	---	--	----------------------------------	---

2.2 Majuscule ou minuscule

Les instructions Python s'écrivent en minuscule. On peut utiliser les majuscules dans les noms des variables, mais attention à bien respecter l'écriture que vous avez choisie (on dit « respecter la casse »)

2.3 Le rôle du symbole « = »

- Le symbole « = » n'est pas celui de l'égalité mais celui de **l'affectation d'une valeur à une variable**. On stocke une valeur numérique ou du texte dans une mémoire.
- La syntaxe est **NomVariable = valeur**
- **Pour afficher la valeur d'une variable : print(NomVariable)**
- Par contre, le symbole « == » teste si une égalité est vraie ou fausse

Exemple : Ouvrir Idle pour Python v3 et tester, dans l'ordre, les lignes de code suivantes

1	2	3	4
>>>a=21 >>>print(a) >>>a	>>>a==20	>>>a==21	>>>b=5 >>>b=b+1 >>> print(b)

2.4 Affectations multiples

```
>>> a,b=10,22
>>> a,b= a+b,a-b
>>>print(a,b)
```

2.5 Exercices

>>> 3+2	>>> s=12 >>> m=24 >>> s=s+m >>> 3*s	>>> s=5 >>> s**2 >>> s**3	>>> n=13 >>> n/2	>>>n//2	>>> n%2
---------	--	---------------------------------	---------------------	---------	---------

3 Travail dans l'éditeur d'Idle

Même pour les petits programmes, il est plus simple de conserver les lignes de code écrites en utilisant l'éditeur fourni avec Idle. Faire :

- **< File > - [New Window]**
- Ecrire le programme dans la fenêtre qui s'ouvre.
- Sauvegarder dans le répertoire de son choix (Attention : ne pas oublier « .py ») : « 31_mon premier programme.py »
- **< Run > - [Run Module]** pour tester

```
# Mon premier programme

# Déclaration des variables
s = 12
m = 24

# Calculs
print ("Somme : ", s + m)    # Affiche la somme
print ("Produit :", s*m)    # Affiche le produit
```

3.1 Chaînes de caractères

Quand on veut définir une chaîne de caractère, on la délimite par des guillemets " ou par une apostrophe '. Pour afficher une chaîne de caractère, il faut utiliser la commande print("Chainedecaractère")

Exemple :

n=5 print(" le carré de ",n,"est",n**2)	abc= "deux mots" print (abc) print (" abc ")
--	--

3.2 Demande d'un nombre

La commande **input()** permet d'entrer une valeur au clavier mais renvoie une chaîne de caractères. Donc si c'est un nombre que l'on veut manipuler, il faudra utiliser, en plus, la fonction **int()** s'il s'agit d'un entier ou **float()** s'il s'agit d'un décimal

```
print("entrer un nombre positif")
chaine=input()
nb=float(chaine)
if nb >0 :
    print("ce nombre convient")
```

La variable chaine prend la valeur entrée par l'utilisateur (sous forme de chaîne de caractère)

La variable nombre transforme la chaîne de caractère en nombre réel

3.3 Test et condition

Si alors ...	Si ... alors ...sinon...	Si ... alors ... sinon si ... sinon ...
if test : <i>Bloc d'instructions si la condition est réalisée</i> <i>Remarque :</i> Le "alors" n'apparaît pas en Python, c'est l'indentation qui délimite le bloc à exécuter.	if test : <i>Bloc d'instructions si la condition est réalisée</i> else : <i>Bloc d'instructions si la condition n'est pas réalisée</i>	if test1 : <i>Bloc d'instructions si la condition au test 1 est réalisée</i> elif test 2 : <i>Bloc d'instructions si la condition au test 2 est réalisée</i> else : <i>Bloc d'instructions si aucune condition n'est réalisée</i>

3.4 Exercices (Ecrire et tester)

1. Ecrire et tester les lignes de code suivantes

```
# Premier exemple de condition
a = 5
if a > 0: # Si a est supérieur à 0
    print("a est supérieur à 0.")
```

```
if a > 0: # Positif
    print("a est positif.")
elif a < 0: # Négatif
    print("a est négatif.")
else: # Nul
    print("a est nul.")
```

```
age = 21
if age >= 18: # Si age est supérieur ou égal à 18
    print("Vous êtes majeur.")
else: # Sinon (age inférieur à 18)
    print("Vous êtes mineur.")
```

2. Cotisation club de sport.

Emilie souhaite s'inscrire dans un club de sport.

Le club propose deux tarifs différents :

- le tarif A : une cotisation annuelle de 80€ et la séance coûte 10,50€;
- le tarif B : la séance coûte 17€.

Voici un programme incomplet qui, suivant le nombre n de séances suivies par Emilie dans l'année, affiche le tarif le plus avantageux :

```
1 n = int(input("Entrer n : "))
2 x = 80 + 10.5*n
3 y = 17*n
4 if x<y :
5     print("Le tarif ... est le plus avantageux")
6 else :
7     print("Le tarif ... est le plus avantageux")
```

1. Que représente la variable x dans le contexte de l'exercice ?
2. Que représente la variable y dans le contexte de l'exercice ?
3. Compléter les pointillés des lignes 5 et 7.
4. Saisir le programme pour compléter le tableau suivant :

n	5	10	12	13	18	20
Tarif le plus avantageux						

3. Location voiture

Dans une agence de location, le coût journalier pour une voiture s'élève à 60€, plus :

- 0,35€ par km parcouru jusqu'à 100 km ;
- 0,45€ par km parcouru au-delà.

1. Vérifier par un calcul que le coût de location pour 160 km parcouru est de 122€.
2. Saisir le programme suivant en le complétant. Quel est le coût pour 375 km ?

```
1 x = float(input("Entrer le nombres de km : "))
2 if x<=100 :
3     c = .....
4 else :
5     c = .....
6 print("Coût de la location : ", c)
```

4. Comparaison des puissances d'un nombre

Saisir et compléter le programme suivant, qui doit déterminer le plus grand nombre entre x^2 et x^3 .

```
1 x = float(input("x = "))
2 y = x**2
3 z = .....
4 if ..... :
5     print("le plus grand nombre est ... qui vaut : ", y)
6 else :
7     print("le plus grand nombre est ... qui vaut : ", z)
```

Compléter le tableau à l'aide du programme :

x	-0,3	8,7	1,2	0,6	-0,01	0,01	1,001
Plus grand nb entre x^2 et x^3							

5. Un programme qui demande un nombre et qui affiche son opposé.
6. Un programme qui demande à l'utilisateur son année de naissance et qui affiche son âge.
7. Un programme qui permet de saisir le nom de l'utilisateur et de renvoyer « Bonjour » suivi du nom.
8. Un programme qui teste si un nombre donné est pair ou impair.

Note : Le reste de la division entière par 2 est donné par : %2