

ACTIVITÉ : PYTHON

PYTHON est un langage de programmation, placé sous licence libre (licence BSD). La première version date de 1990, et son créateur et leader est Guido VAN ROSSUM, grand fan de la série télévisée *Monty Python's Flying Circus*. Sa conception a été notamment influencé par les langages C, Java, Perl et Tcl.

- C'est un logiciel multi-plateforme : de Linux à Windows, en passant par macOS mais aussi de Android à iOS (sur smartphone : Jailbreak (iOS), SL4A (Android)).
- Foultitude d'entreprises travaillent avec ; le langage dispose d'une syntaxe épurée.
- Le langage a une syntaxe mettant de côté les mécanismes de « bas niveau » : on n'interagit pas directement avec les fonctions du processeur (registre, adresse mémoire et autres joyeusetés).

1 Premières commandes

↪ Exemple 1

```

2+3          -4+7          2**3          3/4
2+8*5        2*2          2*4          3//4          45%6
(5+1)*5      2**2        2**4          45/6          45//6+45%6
7-4          2*3          10**3        45//6

```

▣ Définition 1

Somme de a et b	$a+b$	« plus »
Différence de a avec b	$a-b$	« tiret court »
Produit de a et b	$a*b$	« astérisque »
a exposant b	$a**b$	« double astérisque »
Quotient (décimal) de a par b	a/b	« barre oblique » (slash)
Quotient par division euclidienne de de a par b	$a//b$	« double barre oblique » (double slash)
Reste par division euclidienne de de a par b	$a\%b$	« pourcentage »

2 Bibliothèque numpy

Python ne connaît pas de base certaines fonctions mathématiques comme la racine carrée, le cosinus, etc. : on peut les charger en faisant

```

import numpy
numpy.sqrt(5)
numpy.cos(180/numpy.pi*30)

```

N.B. :

- `sqrt` est une abréviation de « square root » ;
- Pour utiliser `cos`, `sin` et `tan` avec des degrés, on utilise l'écriture ci-contre.
- `numpy rint(a)` donne l'arrondi à l'entier près de a ;
- `numpy floor(a)` donne la partie entière (arrondi à l'entier près par défaut) de a ;
- `numpy round(a,n)` donne l'arrondi à 10^{-n} près de a.

3 Typage

↳ Exemple 2

```

type(6)           6 == 6.0           'a'           'l\' apostrophe'
type(6.0)         6 is 6.0           "a"           'str'+ 'chaîne'
6 == 2+4          a                   type('a')     2*'élève'

```

▣ Définition 2

Dans un programme informatique, un ordinateur fait intervenir des **variables informatiques** : elle stocke des valeurs dans un emplacement de mémoire. On s'intéressera à trois **types** de variable

- le type **entier** (integer) qui correspond aux entiers relatifs ;
- le type **flottant** (float), parfois représenté en notation scientifique : $M \times 10^e$ (une mantisse M , nombre décimal compris entre -10 et 10 , et un exposant entier e) qui correspond aux nombres décimaux ;
- le type **chaîne de caractère** (string), qui permet d'implémenter des messages textuelles.

4 Affectation

↳ Exemple 3

```

a=7           b=10           b           n,d=7,4
a             b+=1           b-a           n/d,n*d
a=a+1         b             b-=1           a=a**2-7*b+5
a             b*=2           b

```

▣ Définition 3

- **variable = valeur** permet d'affecter à une variable une valeur.
- On peut affecter à une variable a une valeur dépendant de a et d'autres variables.
- Quelques affectations rapides

Augmenter la variable a de la valeur v	$a += v$
Diminuer la variable a de la valeur v	$a -= v$
Multiplier la variable a par la valeur v	$a *= v$
Diviser la variable a par la valeur v	$a /= v$
Mettre la variable a à l'exposant v	$a **= v$

5 Objet intégré input

↳ Exemple 4

```

nb=int(input('Taper un nombre '))      err=input('Taper un nombre ')
Nb=int(input('Taper un autre nombre ')) err
nb,10*Nb                               type(err)

flt=float(input('Taper un nombre décimal non entier '))
Flt=int(input('Taper un nombre décimal non entier '))
flt, Flt

Tps=str(input('Il fait beau?(O/N) '))
tel=input('Marque de votre téléphone portable : ')
Rép=input('La moitié du cours s\'est écoulé?(V/F) ')
Tps,tel,Rép

```

▣ Définition 4

L'utilisateur peut choisir la valeur d'une variable grâce à l'objet intégré `input()`.
Par défaut, cette variable est traité en chaîne de caractère, mais on peut forcer son typage.

6 Fonctions

↳ Exemple 5

```

def f(x):                               def g(x):                               def cube(x):
    return 4*x-7                         return x**2-x-1                         return x**3

f(0)                                     g(0)                                     cube(0)
f(4)                                     g(4)                                     cube(4)
f(-6)                                    g(-6)                                    cube(-6)

```

▣ Définition 5

On définit une fonction sur PYTHON grâce au mot clef `def`, présentée ainsi

```

def nomfonction(ventrée):
    instructions
    return image

```

On peut alors calculer l'image de entrée grâce à `nomfonction(entrée)`.

Hormis la ligne contenant le mot clef `def`, toutes les lignes servant à définir la fonction commencent par une tabulation : on appelle cet espacement **l'indentation**.

Si l'indentation n'est pas faite, les lignes ne seront pas prises en compte dans la définition de la fonction (et cela peut entraîner des erreurs).

↳ Exemple 6

Une fonction peut avoir plusieurs variables d'entrée.

```

def aire_trapeze(petite_base,grande_base,hauteur):
    return (petite_base+grande_base)*hauteur/2

```

7 Application

→ Exercice 1

Réaliser les calculs suivants à l'aide de Python.

$$4 \times (3 + 5)$$

$$8 + \frac{3}{4}$$

$$\frac{8+3}{4}$$

$$4 - \frac{1}{6 \times 4}$$

$$\frac{5}{14-5} + 5$$

$$3 \times 4^2$$

$$12^3 - 67$$

$$7^3 \times \frac{3^4}{2^5}$$

$$5\sqrt{17-9} + 2$$

$$\frac{1}{\sqrt{3}-4}$$

→ Exercice 2

1. Proposer un module Python qui demande une valeur à l'utilisateur, et affiche le carré de cette valeur auquel on a retranché 10.
2. Proposer un module Python qui demande une valeur à l'utilisateur, et affiche cette valeur augmentée de 40 %.
3. Proposer un module Python qui demande une valeur à l'utilisateur, et affiche cette valeur diminuée de 15,5 %.
4. Proposer un module Python qui demande une valeur à l'utilisateur, et affiche la somme de cette valeur, de son carré et de son cube.

→ Exercice 3

Conjecturer le rôle de ce module Python.

```

valeur=int(input('Quel nombre ? '))
mot=str(input('Quel mot ? '))
print(valeur*mot)

```

→ Exercice 4

Définir sur Python les fonctions suivantes et calculer les images de 7; -2 et $\sqrt{11}$ pour chacune d'entre elles; si la valeur n'est pas exacte, faire aussi afficher un arrondi à 10^{-3} près.

$$f(x) = x^3 - x - 1$$

$$g(x) = \frac{1}{3} - \sqrt{5}x$$

$$h(x) = \frac{6x - 10}{3x^2 + 1}$$

$$k(x) = (x + 1)^4 - (x - 1)^4$$

→ Exercice 5

1. Définir une fonction `Plus_20_pc`, qui prenne en entrée un nombre `valeur` et qui donne en sortie ce nombre augmenté de 20 %.
2. Définir une fonction `Moins_15_pc`, qui prenne en entrée un nombre `valeur` et qui donne en sortie ce nombre diminué de 15 %.
3. Définir une fonction `Evo_rel`, qui prenne en entrée deux nombres `taux` et `valeur` et qui donne en sortie le nombre `valeur` après évolution au taux `taux`.