

# Expressions, variables et fonctions

## I. Les nombres et les opérations

◊ Tout nombre comportant un point décimal ou donné en notation scientifique est considéré comme un *float* (approximation d'un nombre réel). Dans le cas contraire, le nombre est entier (*int*).

```
>>> 0.001
0.001
>>> 2.2e-3
0.0022
```

```
>>> 1e3
1000.0
>>> type(1e3)
<class 'float'>
```

```
>>> 12345
12345
>>> type(12345)
<class 'int'>
```

◊ On a les opérations usuelles +, -, \* et **abs** (valeur absolue).

⚠ La puissance est \*\* et pas ^.

```
>>> 10**2
100
>>> 10^2
8
```

⚠ La division / est la division usuelle. L'opération // renvoie le quotient de la *division euclidienne* et l'opération % renvoie le reste.

```
>>> 3.5/2.5
1.4
>>> 19/2
9.5
```

```
>>> 7/2
3.5
>>> 7//2
3
```

```
>>> 9//4
2
>>> 9%4
1
```

## II. Les variables

◊ L'utilisation de variables est nécessaire dans la plupart des programmes. Une variable permet de faire référence à une valeur au moyen d'un nom. Il faut garder à l'esprit l'image

suivante : la mémoire de l'ordinateur est constituée de boîtes sur lesquelles sont collées des étiquettes et qui contiennent des données.

- ◊ Une variable est décrite par 3 éléments :
  - Son *nom* : c'est l'étiquette de la boîte, c'est le moyen par lequel on peut faire référence à la valeur contenue dans cette boîte;
  - Sa *valeur* : c'est la donnée contenue dans la boîte;
  - Son *type* : c'est l'ensemble auquel appartient la valeur associée à cette variable. Pour l'instant, on ne connaît que 2 types : *int* et *float*. Le type d'une variable conditionne les opérations que l'on peut réaliser avec cette variable. C'est PYTHON qui détermine le type d'une variable en fonction de sa valeur.

```
>>> x = 2
>>> print("Valeur de x :", x)
Valeur de x : 2
>>> y = x/100
>>> print("Valeur de y :", y)
Valeur de y : 0.02
>>> x = x+1
>>> print("Valeur de x :", x)
Valeur de x : 3
>>> print("Valeur de y :", y)
Valeur de y : 0.02
>>> type(x)
<class 'int'>
>>> type(y)
<class 'float'>
```

⚠ L'opération d'affectation (réalisée par le symbole =) permet de donner une valeur à une variable. C'est une erreur grave d'utiliser dans un programme une variable sans lui avoir auparavant donné une valeur.

◊ Les noms de variable que l'on utilisera commenceront systématiquement par une lettre majuscule ou minuscule (**a** . . **zA** . . **Z**) suivie éventuellement d'autres lettres, ou de chiffres (**0** . . **9**) ou du caractère blanc souligné (**\_**, *underscore*).

⚠ Pour rendre les programmes lisibles, il est *impératif* d'utiliser des nombres de variables explicites (par exemple **altitude** plutôt que **h** pour désigner une altitude). Noter que PYTHON fait la différence entre majuscule et minuscule.

**Remarque.** Dans l'interface de PYZO, la fenêtre *Workspace* permet de connaître les variables qui sont actuellement définies dans la console (on retrouve les 3 informations *nom*, *type*, *valeur*). □

## III. Les fonctions (introduction)

◊ Comme en mathématiques, on peut définir des *fonctions* qui reçoivent des *paramètres* et produisent un *résultat*.

```
import math
def volume_cylindre(rayon, hauteur):
    base = math.pi*rayon**2
    return hauteur*base
print(volume_cylindre(1,2))
```

⚠ Le concept de fonction est essentiel et présente de nombreuses subtilités (on lui consacra une séance complète).

6.283185307179586

## Exercices pour la semaine prochaine

### Question 1.

- Définir une variable  $x$  égale à 1.
- Définir une variable  $y$  égale à  $x + 1$ .
- Ajouter 1 à  $x$ .
- Faire afficher  $x$  et  $y$ .

**Question 2.** Faire afficher les valeurs de  $\ln(e)$ ,  $\sin(\pi)$  et  $\sqrt{2}$ .

### Question 3.

- Définir deux variables :  $a$  égale à  $2 \cdot 10^2$  et  $b$  égale à 2.
- Définir une variable  $c$  égale à  $\frac{a}{2b}$  et une variable  $d$  égale à  $b^{1/2}$ .
- Faire afficher  $c$  et  $d$ .

### Question 4.

- Définir une fonction **moyenne** (**a**, **b**, **c**) qui calcule et renvoie les moyenne des trois nombres  $a$ ,  $b$  et  $c$ .
- Utiliser cette fonction pour faire afficher la moyenne des nombres 1, 7 et 4.

## Corrections

### Q1.

```
x = 1
y = x+1
x = x+1
print(x)
print(y)
```

2 2

### Q2.

```
from math import log, e, sin, pi, sqrt

print(log(e))
```

1.0

```
print(sin(pi))
```

1.2246467991473532e-16

```
print(sqrt(2))
```

1.4142135623730951

### Q3.

```
a = 2e2
b = 2
c = a/(2*b)
d = b**(1/2)
print(c)
```

50.0

```
print(d)
```

1.4142135623730951

### Q4.

```
def moyenne(a,b,c):
    return (a+b+c)/3

print(moyenne(1,7,4))
```

4.0