

Séance 6 : compléments sur les types int, float et bool

1°/ Codage des nombres en machine

En informatique, les nombres réels n'existent pas ! Il faut en effet coder tous les nombres dans un langage compréhensible par les ordinateurs. Chaque nombre est ainsi codé par une liste finie de deux chiffres 0 et 1 (on parle de l'écriture des nombres en binaire).
Aucun problème a priori pour les entiers qui ont une écriture en binaire finie.
Python code même de façon exacte de très grands entiers.
Pour les non-entiers, c'est plus compliqué !
Certains décimaux comme 0,5 sont codés de façon exacte en Python. Mais d'autres qui paraissent « simples » dans notre écriture décimale comme 0,1 ne sont pas codés de façon exacte. C'est donc une valeur approchée du nombre qui est codée en machine.

Si on tape dans le Shell : $0.5 + 0.5$ on obtient 1.0
Si on tape dans le Shell : $0.1 + 0.7$ on obtient 0.7999999999999999

Ce n'est ni un bug de Python ni un bug de l'ordinateur !
N. B. : sur la Casio 90+E, on obtient bien 0,8 ...

En résumé : Python travaille avec deux types de nombres :
- les entiers de type **int** (integer) qui sont codés de façon exacte ;
- les autres nombres de type **float** qui sont pour beaucoup codés en machine de façon approchée.
Les calculs sur les entiers uniquement sont des calculs exacts alors que des calculs avec des flottants sont souvent approchés.

Attention : certaines opérations produisent des flottants, même si, d'un point de vue mathématiques, le résultat est un entier. C'est le cas de la division et de la racine carrée :
si on tape dans le Shell : $(6+4)/2$ on obtient 5.0 (un flottant) et non pas 5
si on tape dans le Shell : $\text{sqrt}(25)$ on obtient 5.0 (un flottant) et non pas 5

5 et 5,0 sont deux écritures d'un même nombre entier en mathématiques ;
5 et 5,0 ne sont pas le même objet en Python.

2°/ Conséquence sur les booléens

Lorsque l'on veut effectuer un test, on crée une expression, par exemple $a > b$ qui peut prendre deux valeurs : Vraie (*True*) ou Faux (*False*) suivant les valeurs de a et b .
On a ainsi créé un booléen.

Pour tester une égalité, le signe d'égalité étant utilisé pour l'affectation, on utilise un « double égal » (comme nous l'avons vu à la *séance 3*).

Les deux égalités mathématiques $3^2 + 4^2 = 5^2$ et $\left(\frac{3}{7}\right)^2 + \left(\frac{4}{7}\right)^2 = \left(\frac{5}{7}\right)^2$ sont vraies.

Mais : si on tape dans le Shell : $3^2 + 4^2 == 5^2$ on obtient *True* ;
si on tape dans le Shell : $\left(\frac{3}{7}\right)^2 + \left(\frac{4}{7}\right)^2 == \left(\frac{5}{7}\right)^2$ on obtient *False*.

En résumé : On ne peut pas tester de façon certaine une égalité entre deux nombres de type **float**.
Il est donc important de déterminer les types de variables en jeu !